National Library of Canada

Acquisitions and Bibliographic Services Branch

395 Wellington Street
Ottawa, Ontario
K1A 0N4

Bibliothèque nationale du Canada

Direction des acquisitions et des services bibliographiques

395, rue Wellington
Ottawa (Ontario)
K1A 0N4

Your file Votre référence

Our file Notre référence

NOTICE

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Reproduction in full or in part of this microform is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30, and subsequent amendments.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

La reproduction, même partielle, de cette microforme est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30, et ses amendements subséquents.

Canada

UNIVERSITY OF ALBERTA

A
STEREO-VISION SYSTEM
FOR THE
INTRAOPERATIVE MONITORING
OF
SCOLIOSIS CORRECTION

by

Ketan Bhalla

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of Master of Science.

DEPARTMENT OF ELECTRICAL ENGINEERING

Edmonton, Alberta

FALL 1995

National Library
of Canada

Acquisitions and
Bibliographic Services Branch

395 Wellington Street
Ottawa, Ontario
K1A 0N4

Bibliothèque nationale
du Canada

Direction des acquisitions et
des services bibliographiques

395, rue Wellington
Ottawa (Ontario)
K1A 0N4

Your file Votre référence

Our file Notre référence

THE AUTHOR HAS GRANTED AN IRREVOCABLE NON-EXCLUSIVE LICENCE ALLOWING THE NATIONAL LIBRARY OF CANADA TO REPRODUCE, LOAN, DISTRIBUTE OR SELL COPIES OF HIS/HER THESIS BY ANY MEANS AND IN ANY FORM OR FORMAT, MAKING THIS THESIS AVAILABLE TO INTERESTED PERSONS.

L'AUTEUR A ACCORDE UNE LICENCE IRREVOCABLE ET NON EXCLUSIVE PERMETTANT A LA BIBLIOTHEQUE NATIONALE DU CANADA DE REPRODUIRE, PRETER, DISTRIBUER OU VENDRE DES COPIES DE SA THESE DE QUELQUE MANIERE ET SOUS QUELQUE FORME QUE CE SOIT POUR METTRE DES EXEMPLAIRES DE CETTE THESE A LA DISPOSITION DES PERSONNE INTERESSEES.

THE AUTHOR RETAINS OWNERSHIP OF THE COPYRIGHT IN HIS/HER THESIS. NEITHER THE THESIS NOR SUBSTANTIAL EXTRACTS FROM IT MAY BE PRINTED OR OTHERWISE REPRODUCED WITHOUT HIS/HER PERMISSION.

L'AUTEUR CONSERVE LA PROPRIETE DU DROIT D'AUTEUR QUI PROTEGE SA THESE. NI LA THESE NI DES EXTRAITS SUBSTANTIELS DE CELLE-CI NE DOIVENT ETRE IMPRIMES OU AUTREMENT REPRODUITS SANS SON AUTORISATION.

ISBN 0-612-06444-1

Canada

Lippincott - Raven
PUBLISHERS

A Wolters Kluwer Company

August 2, 1995

Ketan Bhalla
Dept. of Electrical Engineering
238 Civil/Electrical Building
Edmonton, Alberta T6G 2G7
CANADA

Dear Ms. Bhalla:

Thank you for your recent request to use material from one of our publications in your thesis.

The quoted material may be included in your manuscript, using the standard format and footnotes suggested in the UNIVERSITYOF CHICAGO STYLE MANUAL or those required by your university.

However, if your thesis is selected for publication and a contractual agreement has been signed, then you should submit your formal permission request to this office. Please advise the name of your publisher, tentative publication date, number of pages in your forthcoming book and the estimated retail price. Upon receipt of this information this office shall then research your request and respond with the conditions of the permission.

This course of action must be taken since many times representation of the copyrighted material may change between the time a thesis is submitted and the date that a contractual arrangement for publication has been secured.

Congratulations as you complete your advanced studies, and with very best wishes for your future work!

Sincerely,

Marie P. Wayne

Marie P. Wayne
Permissions Editor

/T

227 East Washington Square, Philadelphia, PA 19106-3780 • Tel: 215-238-4361 • Fax: 215-238-4419

University of Alberta

Library Release Form

Name of Author:	Ketan Bhalla
Title of Thesis:	A Stereo-Vision System for the Intraoperative Monitoring of Scoliosis Correction
Degree:	Master of Science
Year this Degree Granted:	1995

Permission is hereby granted to the University of Alberta Library to reproduce single copies of this thesis and to lend or sell such copies for private, scholarly, or scientific research purposes only.

The author reserves all other publication and other rights in association with the copyright in the thesis, and except as hereinbefore provided, neither the thesis nor any substantial portion thereof may be printed or otherwise reproduced in any material form whatever without the author's prior written permission.

Ketan Bhalla

(Student's Signature)
Ketan Bhalla,
L-20, South Extension, Part 2,
New Delhi 110 049.
India.

Date: Aug. 4, 1995

University of Alberta

Faculty of Graduate Studies and Research

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research for acceptance, a thesis entitled '*A Stereo-Vision System for the Intraoperative Monitoring of Scoliosis Correction*' submitted by *Ketan Bhalla* in partial fulfillment of the requirements for the degree of *Master of Science*.

N.G. Durdle

A.E. Peterson

S. Bamforth

V.J. Raso

K.A. Stromsmoe

Date: Aug. 1/95

To my parents, Lalit and Kusum Bhalla.

In memory of Dada and Nana.

Abstract

A stereo-vision system was developed to automatically track the three dimensional (3D) motion of artificial markers which were rigidly attached to vertebrae during the surgical correction of scoliosis. The trajectory of these markers can be interpreted in terms of the 3D dynamic configuration of the spine to provide visual and quantitative feedback for different stages of surgery.

Development addressed all issues of the feature-based stereo-vision paradigm. Images were enhanced. Feature detection was implemented as a template-matching problem with multiple templates. Similarity measures were analyzed for correlation of gray level and edge images. Correspondence used scene and system constraints to register features extracted from the stereo-images. Triad-markers were identified in the reconstructed 3D feature space by model-matching. Experiments and results are provided with operating room image data for all the work presented.

Acknowledgements

Like any long, multi-disciplinary piece of work, this project owes its success to a number of people. I would like to express my gratitude to:

my supervisors, Dr. Nelson G. Durdle and Dr. Art E. Peterson for their encouragement, support, and guidance throughout this project.

surgeons, Dr. Jim Mahood and Dr. Mark Moreau, for being so accommodating and helpful through the slow experimentation and development processes. The financial assistance they arranged for is much appreciated.

Jim Raso and Doug Hill of the Glenrose Rehabilitation Technology Group for their valuable guidance with research, assistance with the drafting of papers and the thesis. I am obliged to Doug for all his help - from course projects to the operating room to dropping me back to University at the slightest opportunity !

Dr. X. Li of the Department of Computing Science for providing helpful insights into template matching for the solution of the *detection problem.*

Herb Ganz of the EE Machine Shop for his tremendous effort on the mechanical development of the system and triad markers.

my friends: Mark, Roland, Mike, Richard, and Amar for the many discussions on the system responsible for leading (and misleading) me on so many occasions.

the ladies at the department office - Michelle, Carla, Shirley, and Corina - thanks for all the help with keys, printing, faxing, and for all those colorful discussions.

Norman's stereo, Edmonton's FM radio stations, olson and angelo (workstations), Mark's reliable K-car, the Rockies (especially, Mount Hawk and Roche Miette), my two goldfish, Canadian beer, BITSian wingies, my roommates (Apoorv, Andrew, and Yao), Archana (for those bittersweet 9 weeks), friends from the International Center, and the many other (living and non-living) entities (in no particular order) involved, directly or indirectly, with the life/work of the author.

Mummy, Papa and Shalu for their undying love and encouragement. Nilu Bua and Anil Uncle for all their trust and assistance.

For everyone and everything, I am, and will always remain, one extremely grateful person.

Contents

1 Introduction **1**

1.1 Purpose 1

1.2 Motivation 1

1.3 Overview 2

2 Background **4**

2.1 Scoliosis 4

2.2 Intraoperative Monitoring 6

2.3 Stereo-vision and the OR 9

2.3.1 Image Acquisition 10

2.3.2 Photogrammetry 12

2.3.3 Image Processing and Computer Vision 12

2.3.4 Visualization 13

3 3D Perception **15**

3.1 Concepts and Methods 15

3.2 Camera Calibration 19

3.2.1 DLT 20

3.2.2 Collinearity Equations 20

3.3 Results 22

4 Feature Recognition and Low Level Processing **29**

4.1 Analysis of Operating Room Images 30

4.2 Preprocessing 33

4.2.1 Gamma Correction 33

4.2.2 Automatic Size Reduction 34

4.2.3 Mode Filtering 37

4.2.4 Edge Detection 45

4.3 Model-Based Recognition 48

4.3.1 Similarity Measures 49

4.3.2 Template Design and Organization 56
4.3.3 Feature Extraction 65
4.3.4 Results 69

5 Hierarchical Feature Correspondence **72**
5.1 Literature Review 73
5.2 Correspondence Strategy 74
5.3 Candidate Sets 75
5.4 Refinement of Candidate Sets 82
5.5 3D Feature Reconstruction 87
5.6 Triad-marker Identification 89
5.7 Triad Representation and Results 93

6 Visualization **103**
6.1 Clinical Requirements 103
6.2 Graphical Representation 105
6.3 Outstanding Issues 106

7 Conclusions and Recommendations **111**
7.1 Summary of Results 111
7.2 Recommendations for Future Work 113
7.3 Conclusions 119

A Correlation Speedup using FFTs **128**

B System Specifications **134**

C Software Model: DETECT-POINT **135**
C.1 Module Descriptions 136

D Software Model: MATCH-POINT **139**
D.1 Module Descriptions 140

List of Tables

3.1 System Equipment and Camera Geometry. 23

3.2 Control Field Coordinates. 25

3.3 Calibration Parameters for trial C. 27

3.4 Reconstruction Accuracy using trial C parameters. The standard deviation of residuals is quoted for the reconstruction of each trial. The mean of residues was zero in each case. 27

4.1 Selectivities of Measures. 55

4.2 Proximity Matrix for the 6 Gray Level-Space Representative Templates. 64

4.3 Proximity Matrix for the 4 Edge-Space Representative Templates. . . 64

4.4 Manual versus Automatic Detection. 71

5.1 Structural Similarity with a small baseline-vergence configuration. . . 84

5.2 Structural Similarity with a large baseline-vergence configuration. . . 85

5.3 Distance and error matrices. 91

5.4 Triad matching with a small baseline-vergence configuration. 100

5.5 Triad matching with a small baseline-vergence configuration (contd.). 101

5.6 Triad matching with a large baseline-vergence configuration. 102

6.1 Triad information for both cases. 109

List of Figures

2.1 'Normal' Back Surface and 'Scoliotic' Back Surface. 5

2.2 Pre- and Postoperative radiographs. 5

2.3 Attachment of instrumentation rods during surgery. 7

2.4 Scoliosis Correction using CD Instrumentation. 9

2.5 Model of the OR System and its Components. 10

2.6 Mobile platform with stereo-rig, computer, digitizer, and monitor. . . 11

3.1 Perspective Projection with Stereo Geometry. 16

3.2 The Collinearity Condition : Convention and Terminology. 21

3.3 The *Peg Board*. 24

4.1 A triad-marker model. 29

4.2 Contrast Adjusted Operating Room Image (Left Camera). 31

4.3 Contrast Adjusted Operating Room Image (Right Camera). 31

4.4 Histogram of Preprocessed Right OR Image. 32

4.5 Raw Operating Room Image. 35

4.6 Image after Gamma Correction, iteration #1 ($\gamma = 2.0$). 35

4.7 Image after Gamma Correction, iteration #2 ($\gamma = 1.5$). 36

4.8 Distribution of Averages of Image Segments for fig. 4.7. 38

4.9 Averages distribution (fig. 4.8) after thresholding. 38

4.10 Distribution of Standard Deviations of Image segments for fig. 4.7. . . 39

4.11 Standard deviation distribution (fig. 4.10) after thresholding. 39

4.12 Original OR Image. 40

4.13 Automatically detected surgical field of view for fig. 4.12. 40

4.14 Typical local intensity histogram of a 3x3 pixels neighbourhood. . . . 42

4.15 Continuous analogue of fig. 4.14. 42

4.16 Rationale for truncation in TMF operation. 43

4.17 A high contrast image before median filtering. 44

4.18 The image (fig. 4.17) after truncated median filtering. 44

4.19 Zoomed portion of a low contrast image containing a triad. 44

4.20 The edge image of fig. 4.19. 46
4.21 Fig. 4.19 processed by the 3x3 TMF. 46
4.22 The edge image of fig. 4.21. 46
4.23 3x3 Sobel Edge Masks for the x, y directions 47
4.24 OR image in edge-space. 47
4.25 Gray level test image. 53
4.26 Correlation of the gray level image (fig. 4.25) using the measure ED. . 53
4.27 Correlation of the gray level image (fig. 4.25) using the measure OCC. 54
4.28 Correlation of the gray level image (fig. 4.25) using the measure NORM. 54
4.29 Correlation of the gray level image (fig. 4.25) using the measure MEANS. 54
4.30 Correlation of the gray level image (fig. 4.25) using the measure RHO. 54
4.31 Average Responses of different measures using gray level templates. . 55
4.32 The edge image corresponding to fig. 4.25. 56
4.33 Correlation of the edge image (fig. 4.32) using the measure ED. . . . 57
4.34 Correlation of the edge image (fig. 4.32) using the measure OCC. . . 57
4.35 Correlation of the edge image (fig. 4.32) using the measure NORM. . 57
4.36 Correlation of the edge image (fig. 4.32) using the measure MEANS. . 57
4.37 Correlation of the edge image (fig. 4.32) using the measure RHO. . . 58
4.38 Average Response of different measures using edge templates. 58
4.39 Grey Level-Space Template Tree. 60
4.40 GL template tree samples from left image. 62
4.41 GL template tree samples from right image. 62
4.42 Edge-Space Template Tree. 63
4.43 The Gray level Property Space. 66
4.44 Decision Space corresponding correlation of fig. 4.43. 66
4.45 The Edge Magnitude Property Space. 67
4.46 Decision Space corresponding correlation of fig. 4.45. 67
4.47 Feature Space after dynamic thresholding (K=2, 6 pixels/component). 69
5.1 Hierarchical Correspondence Paradigm for the OR system. 76
5.2 Determining the epipolar segment. 78

5.3 Calculation of 3D close and far points. 79
5.4 Perpendicular distance from the epipolar line segment. 82
5.5 Space Intersection. 88
5.6 A flat, planar triad model. 90
5.7 Calculation of triad position and orientation. 94
5.8 Left stereo-image with annotated features (F for false positives). . . . 96
5.9 Right stereo-image with annotated features (F for false positives). . . 96
5.10 Left image feature #4. 97
5.11 Correspondence Stage I for LIF#4. 97
5.12 Correspondence Stage II for LIF#4. 98
5.13 Correspondence Stage III for LIF#4. 98
6.1 Body and triad vectors for the *ith* vertebra. 104
6.2 SPINE: Pre-correction Configuration. 107
6.3 SPINE: Post-correction Configuration. 108
C.1 DETECT-POINT: Software Modules. 135
D.1 MATCH-POINT: Software Modules. 139

1 Introduction

Appearances to the mind are of four kinds.
Things either are what they appear to be;
or they neither are, nor appear to be;
or they are, and do not appear to be;
or they are not, and yet appear to be.
Rightly to aim in all these cases is the wise main's task.

Discourses, Book 1, Chapter 27,
Epictetus c.50-120

Scoliosis [67] is a deformity of the spine. Scoliotic patients have spines with anomalous curvature profiles and abnormally rotated vertebrae. Surgery is used to correct extreme, progressive deformities. A system has been developed for the three dimensional measurement of corrections made during scoliosis surgery.

1.1 Purpose

This work describes the research and development of a computer vision system to intraoperatively monitor scoliosis surgery. The system was used to make automatic three-dimensional measurements on the displacements and derotations of vertebrae during surgical correction of scoliosis. It can be used to provide surgeons with real-time feedback regarding the changes they are affecting to the spine. It can be used as a tool for monitoring, visualization (using 3D graphical models), assessment, planning, and recording different stages of surgery. Also, unlike other measurement alternatives like radiography, such a system will not pose any radiation hazards.

1.2 Motivation

Traditionally, posterior-anterior (PA) and lateral radiographs have been used for the assessment of three dimensional deformity in scoliosis. Surgical correction is also

assessed using pre- and post-operative x-ray pairs. This method presents several difficulties. The post-operative spine is fastened to surgically implanted stainless steel instrumentation. This instrumentation obstructs various landmarks and features on the spine, preventing any measurements which rely on these landmarks. Standard measures, such as the Cobb angle [15], are essentially two dimensional and do not provide reliable three dimensional information [44]. The use of two dimensional radiographs to visualize three dimensional scenes can be conceptually difficult. As radiographs prove to be an inadequate tool for studying the 3D spatial curvature and rotation of the spine, researchers have been investigating alternative methods [34, 61].

Computer vision was investigated as an alternative to radiographs. It was believed that a vision system based on calibrated stereo-cameras can be used to provide safe, accurate and non-invasive three dimensional information in near real-time. The challenge in this approach lay in the automatic processing of the surgical field-of-view (FOV). As the surgical FOV presents a complex scene, one which is beyond the scope of current computer vision theory, artificial markers were rigidly mounted on vertebrae. The problem was thus altered to the 3D monitoring of these mounted markers. Because these markers conformed to an *a priori* model, they could be reconstructed in three dimensional space. The location and attitude of the markers will be used to three dimensionally model the configuration of the spine and infer the motion of vertebrae to which markers are attached.

1.3 Overview

The computer vision approach presents a number of interesting problems. The design and development of the system was based on photogrammetric principles, image processing, and computer vision techniques. Also, there were considerations due to the fact that the system was to function in the operating room. These and other issues are dealt with in the following chapters.

Chapter 2 provides information on the operating room (OR) problem and the research background on which this work is based. Literature from various fields applicable to this project is discussed. A system description is also provided.

Chapter 3 describes the stereo-camera geometry used to make accurate three dimensional (3D) measurements. It deals with the photogrammetric basis and terminology used during the calibration of the stereo-cameras. Typical calibration and reconstruction results are also presented.

Chapter 4 presents the low level image processing research and implementation for feature detection in OR images. Feature extraction is the phase concerned with the location of artificially planted markers (features) in stereo-images.

Chapter 5 deals with higher level vision processes developed for correspondence of image feature sets between stereo-images. These feature sets were returned by the detection routines of chapter 4. Reconstruction of features with their candidates and consequent model-related inferences in three dimensional space are also covered as part of the correspondence process.

Chapter 6 is a description of the visualization part of the project. Radiographs (x-rays) and camera data were made compatible so that position and orientation changes recorded by the stereo-cameras may be represented in the radiograph frame of reference and portrayed graphically.

Chapter 7 reviews the work done and results obtained. Future work is proposed based on personal experience with the project and its limitations. The thesis closes with conclusions about the project and its potential benefits.

The appendices contain a mathematical formulation, system specifications, and the software model of the system. Documented system software was written in C language on an IBM RS/6000.

2 Background

This chapter introduces the medical background for the technical work described in this thesis. Scoliosis is introduced with some methods of treatment. The need for intraoperative monitoring and the use of stereo-vision for this purpose are justified. An overview of the system and its components is also provided.

2.1 Scoliosis

Scoliosis [67] is characterized by abnormal curvature and axial rotation of the spine (fig. 2.1). The most common form of scoliosis is *idiopathic* scoliosis, meaning 'spinal curvature without a cause'. It is believed [31] that 2-4% of the population have scoliosis curves of at least 10 degrees, and that 0.1-0.3% of children have curvatures greater than 20 degrees. In adolescents, scoliosis can progress at an alarming rate and have severe effects on the cosmesis of the individual. These cosmetic 'defects' include unlevel scapula (shoulders) and hips, a prominent shoulder blade or breast, and poor posture. In those with severe deformities, scoliosis causes fatigue and decreased lung function.

Few measures exist to quantify the degree of deformity or rotation of the scoliotic spine. The Cobb angle (fig. 2.2), which is the angle subtended by the normals to the points of inflexion (maximum curvature) in the spinal midline, is the most popular measure [3, 15]. However, as scoliotic deformity is three dimensional, the Cobb angle alone does not convey complete information about the deformity. The tools most extensively used to visualize deformity are stereo-radiographs (x-rays). The posterior-anterior (PA) and lateral (LAT) radiographs (fig. 2.2) are commonly used to assess deformity, make measurements, plan, and evaluate surgeries [21, 40].

In spite of over a century of research, little is known about idiopathic scoliosis. Consequently, there are no preventive measures to avoid it. Various means of treatment of children with scoliosis have evolved though. These consist of bracing, casting, and corrective surgery [25]. The quantitative 3D correction of the spine affected by each of these methods is still largely unestablished. Surgery is only used on extreme

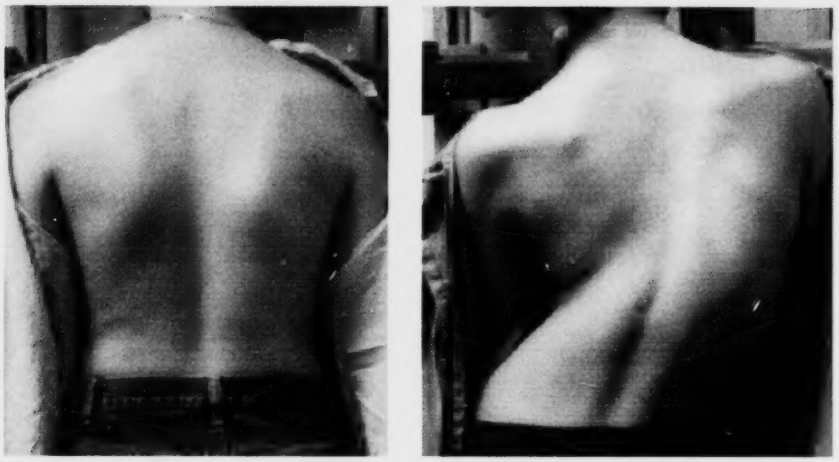

Figure 2.1: *'Normal' Back Surface and 'Scoliotic' Back Surface.* Courtesy: *Rehab. Tech., Glenrose Hospital.*

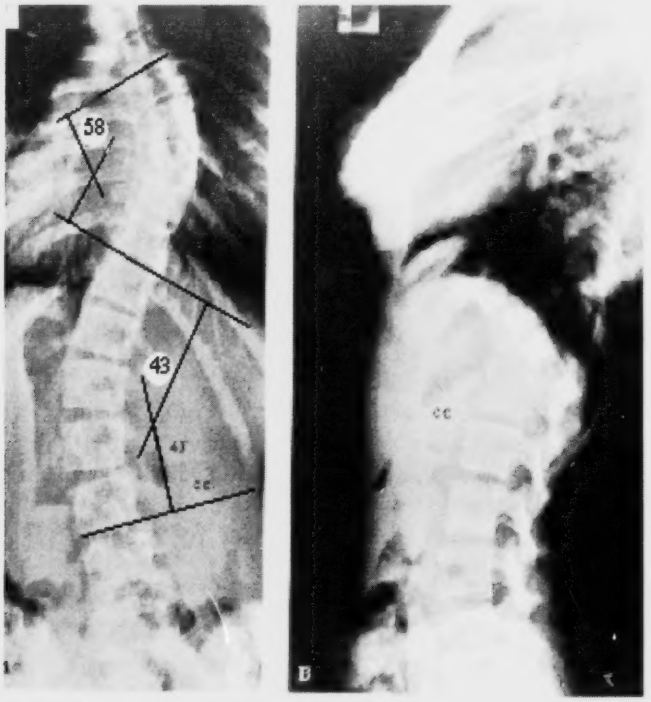

Figure 2.2: *Posterior-Anterior (PA) and lateral (LAT) radiographs. Cobb angles are marked on the PA radiograph.* Courtesy: *Rehab. Tech., Glenrose Hospital.*

curvatures, generally characterized by Cobb angles $> 50^{o}$ [21]. It is invasive and involves the rigid attachment of instrumentation to the spine (fig. 2.3). Appropriate corrective distractive and rotational forces are then applied to the instrumentation to force the spine to attain an improved shape. Advocates of some forms of instrumentation like the Cotrel-Dubousset (CD) believe that correction is mainly the result of the rotation of vertebrae during instrumentation derotational maneuvers [48, 67]. Others, believe in Harrington maneuvers which involve applying stretching or distractive forces to straighten the spine [48]. Currently, it is unclear which philosophy of correction is optimal - distractive or derotative (fig. 2.4 shows a derotative surgery). This ambiguity arises because the conventional tool for scoliosis measurements, radiographs, are not reliable, accurate, or convenient for intraoperative assessment of 3D correction. However, other than radiographs, there are no widely used tools for this purpose.

2.2 Intraoperative Monitoring

Measurements on the scoliotic spine are usually based on radiographs. Though radiographs are ubiquitous, they present many limitations. The procedure for radiograph measurements is cumbersome. It involves the manual detection of vertebral landmarks and features on radiographs. As radiographs do not produce clear images of vertebrae, the positions of these landmarks are ambiguous and consequently, measurements made are prone to errors. The fact that radiographs are non-uniformly distorted also leads to errors in reconstruction of otherwise legitimately identified landmarks. It is even more difficult to make measurements on a spine with instrumentation attached because much of the topology of the vertebrae is obscured by the instrumentation. Also, as measurements are only possible manually and are time consuming, it is not possible to obtain results during surgery. These reasons have caused the investigation of other methods for three dimensional measurements on the spine.

Intraoperative measurements on the spine allow surgeons to closely monitor and record the corrections they make. If accurate and quantitative, intraoperative mea-

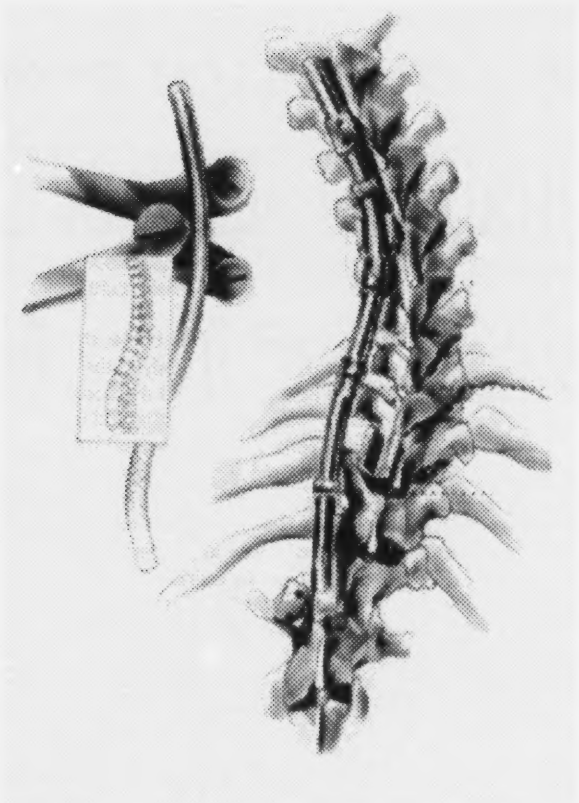

Figure 2.3: *Attachment of the instrumentation rod to the spinal column in a CD surgery. The rod is shaped like the desired spine after derotation. (Courtesy of J.B. Lippencott Inc.)*

surements could be used to determine the correction obtained during each surgical stage. The relative merits of distraction and derotation could then be investigated. The following characteristics of an intraoperative measurement tool for scoliosis were established:

- it must satisfy all operating room constraints of safety and sterility,
- it should be non-invasive, convenient to use, and transparent to surgical procedure,
- it should make automatic on-line measurements and provide easily interpreted results in near real-time,
- it should make accurate measurements in three dimensions (approx. 1-2 mm and 1°),
- it should have the capability to record and present visual results.

Preliminary research in this area has been conducted by Labelle [34] and Treadwell [61]. Labelle *et. al.* used an electromagnetic digitizer to make measurements on previously drilled sites. This involved preparation of the spinal column and the drilling of holes at characteristic sites for the digitizer. This approach is manual and violates the non-invasive requirement of the ideal tool. Other problems [34] include errors due to breathing of the patient during digitization and interference of metallic objects with the electromagnetic fields used by sensors. A *one-shot* measurement process would be more accurate and less distracting to surgeons. Treadwell *et. al.* used traditional surveying techniques with fixed single lens reflex (SLR) cameras. Measurement was an off-line process since photographs had to be manually processed; results from photographs were available only after surgery . This violated the ideal tool requirement of on-line measurements. Such a method is not suitable for real-time monitoring of correction during surgery. Like radiographs, the manual digitization of photographs leaves this method open to errors of judgement.

A new approach which satisfied all the requirements of an intraoperative measurement tool has been implemented. The approach consisted of using calibrated

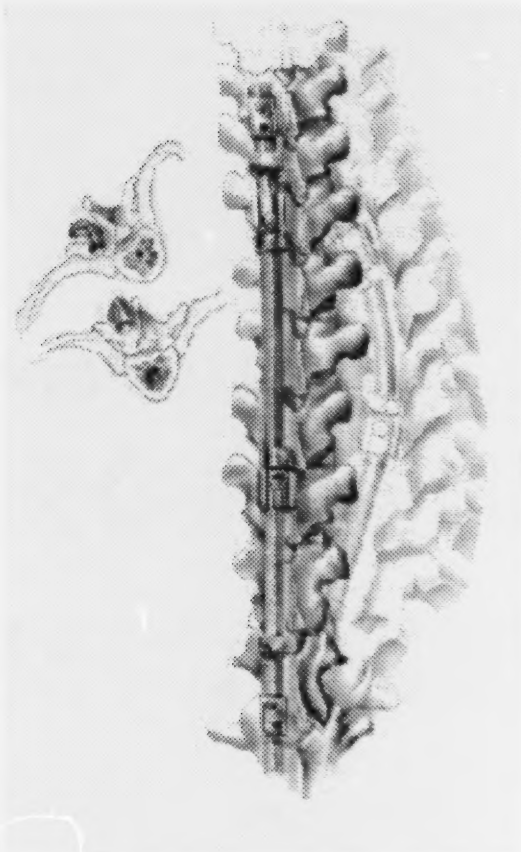

Figure 2.4: *Correction of Scoliosis by a derotational maneuver [34] in a CD instrumentation case. (Courtesy of J.B. Lippencott Inc.)*

stereo-cameras with a digitizer and computer to take intraoperative images during surgery and automatically reconstruct a model of the spine. Reconstruction used camera parameters and returned three dimensional position and orientation information about vertebrae. The technical components of the system are dealt with in the next section.

2.3 Stereo-vision and the OR

The implementation of a stereo-vision system to record intraoperative measurements involved the integration of mechanical design, photogrammetry principles, computer vision and image processing, heuristics, and visualization. In addition, the operating

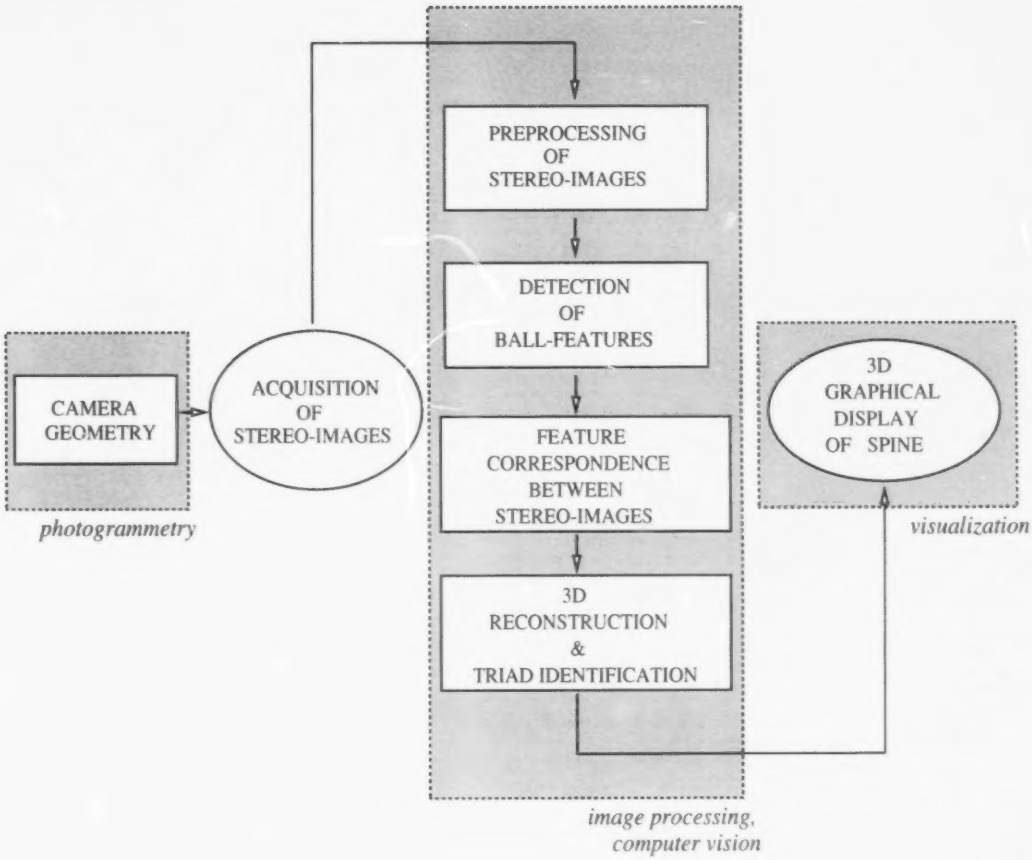

Figure 2.5: *Model of the OR System and its Components.*

room environment placed several constraints on the system.

A functional overview of the system and its components is presented here using the system model (fig. 2.5). The complexity of the surgical FOV was reduced by attaching triad-markers (fig. 4.1) on selected vertebrae. The objective of the image processing and vision algorithms was to automatically locate and orient these markers which represent the vertebrae to which they are attached. The development of each component for the OR-system is briefly discussed in the following subsections.

2.3.1 Image Acquisition

Machine vision systems [12] use fixed cameras to monitor objects of interest. However, in the OR, this proposition could not be applied as the height at which the

triad-markers would reside varied with the height of the surgery-bed. Surgeons were opposed to attaching cameras to improvised OR bed posts as they would prove inconvenient to the support staff in the OR. A mobile platform with a mounted stereo-rig, computer, digitizer and monitor (fig. 2.6) was chosen to be the most flexible approach to image acquisition.

Although the mobile platform did not provide a fixed frame of reference, all measurements of triad-markers were taken relative to a *reference* triad. The *reference* was placed on a vertebrae whose position was expected to be least modified during surgery. Reconstruction results of triad-markers were used relative to this reference triad. This feature allowed comparison of vetebrae between different stages of surgery. Clinically, this feature could be used for the post-operative assessments of the vertebral movements during different stages of surgery.

Figure 2.6: *Mobile platform with stereo-rig, computer, digitizer, and monitor.*

2.3.2 Photogrammetry

Photogrammetric concepts were used to derive an optimal stereo-camera geometry for the system and to establish the calibration parameters of the cameras. There are 13 parameters for each camera, 3 for position, 3 for orientation, 1 for focal length, 2 for principal center coordinates, and 4 for lens distortion. A survey of calibration methods [9] was conducted to choose a suitable method to compute these parameters. The objective was to identify the most accurate method for the calibration of off-the-shelf (non-metric) CCD cameras for the stereo-camera geometry chosen for the OR.

An approach, based on the classical photogrammetry collinearity condition [51], was chosen as the accuracy of this method was excellent (better than 1/700 for the range specified for the OR) and the software and expertise were readily available. Simulations were conducted with the proposed camera geometry to design a control field.

2.3.3 Image Processing and Computer Vision

The operating room (OR) is an unconventional place to apply stereo-vision. Images of surgical fields-of-view (FOVs) were complex when considered for automated analysis. There were few well defined features, the images had regions with high intensities and low entropies (the surgical drape) as well as regions with low intensities and high entropies (surgical FOV). There were imaging problems due to glare from blood, body fluids and instrumentation, and inconsistency in images acquired from one surgery to another. The largest obstacle was simulation of the surgical field-of-view for experimentation with lighting, camera apertures, focii *etc.* As it was not possible to photometrically model the FOV, all testing involved real surgeries instead of simulations. This meant that experimentation was limited within the contraints of the OR for brief periods of time when surgery could be suspended.

The acquisition of stereo-images yields two views of the surgical FOV with markers attached to the spine. Reconstruction in 3D was possible if the location of triad-components was known in both views. This was done automatically using image

processing and computer vision algorithms. This process is often called the stereo paradigm [5, 19]. The main stages of the stereo-paradigm, as implemented for the operating room, are outlined here:

Preprocessing was aimed at transforming raw images into images with characteristics suitable for processing by latter stages. The transforms included contrast enhancement, mode filtering, edge sharpening, edge detection, and identification of the surgical FOV within raw images.

Feature Detection was responsible for returning a set of possible locations of ball-features, the components of triad-markers, in both images. A correlation-based template matching approach was used. An assessment of many correlation measures was also done.

Feature Correspondence was the process of registration of ball-features between stereo-images. It also served to eliminate some *false positives* from the previous detection stage. Registration used multiple constraints like disparity and distance from epipolar segments. The matching algorithm used was not designed to provide unique matches.

Triad Reconstruction was responsible for the calculation of triad-marker locations and orientations in 3-space. The marker model was used to configure 3D points into triads. False triads were eliminated using heuristic rules.

2.3.4 Visualization

Previous research between the University of Alberta and the Glenrose Hospital [63] has developed a GL-based (Graphics Language) graphics visualization software to display a spine model. This was ported to the OpenGL standard and modified for use with triad-data returned by the camera-system [14]. Triads may be attached in any position and orientation relative to the *body* vector representative of a vertebra. The camera system only measures 3D changes in triad-markers. To realistically represent the spine model and record vertebral changes, triad-marker movements have to be

translated into the movements of their corresponding vertebrae. This is only possible if there is some *a priori* knowledge of the *preoperative* configuration of the spine, especially the vertebrae with triad-markers attached. A method selected by Hill and Raso [56] was used to estimate the initial configuration of the spine. Relating this information to the preoperative results of triad positions, triad changes could be translated to vertebral changes. The OpenGL program also has the capability for the graphical display of the spine using 3D vectors instead of vertebra models.

3 3D Perception

The essential function of a camera-based 3D measurement system is the accurate reconstruction of 3D object-space coordinates using 2D image coordinates. One method for 3D perception that is relatively well understood is binocular stereopsis [5] in which two images recorded from different perspectives are used. Intraoperative monitoring used stereopsis for the 3D reconstruction of known entities in both views.

This chapter has three sections. The first introduces basic concepts of stereopsis and cameras. It also briefly describes the investigation of some approaches to modelling a stereo-camera system. The second section provides the background for the method used for camera calibration. The last section describes the camera geometry used and provides results of calibration and reconstruction.

3.1 Concepts and Methods

To realize 3D perception, the transformation from a 3D object-space point to a 2D image point must be understood. This understanding must be extended to a dual-camera system to enable the reverse process of 3D reconstruction from image coordinates. The transformation is based on the modelling of the camera as an instrument which *perspectively* projects points onto the image plane [4, pg. 19]. Perspective projection (fig. 3.1) implies that the image point ($\mathbf{P}_L$) is the intersection of the image plane ($\mathbf{I}_L$) with the line ($\mathbf{PC}_L$) joining the object-space point ($\mathbf{P}$) to the perspective center ($\mathbf{C}_L$).

Concepts

The first concept to be developed here deals with a generalized form of 3D to 2D mapping for a single camera -

- Transform the point from the 3D object-space coordinate system to the 3D camera coordinate system. This involves rotation and translation matrices relating the two coordinate systems.

- Perspectively project the 3D point onto the image plane. This yields the ideal

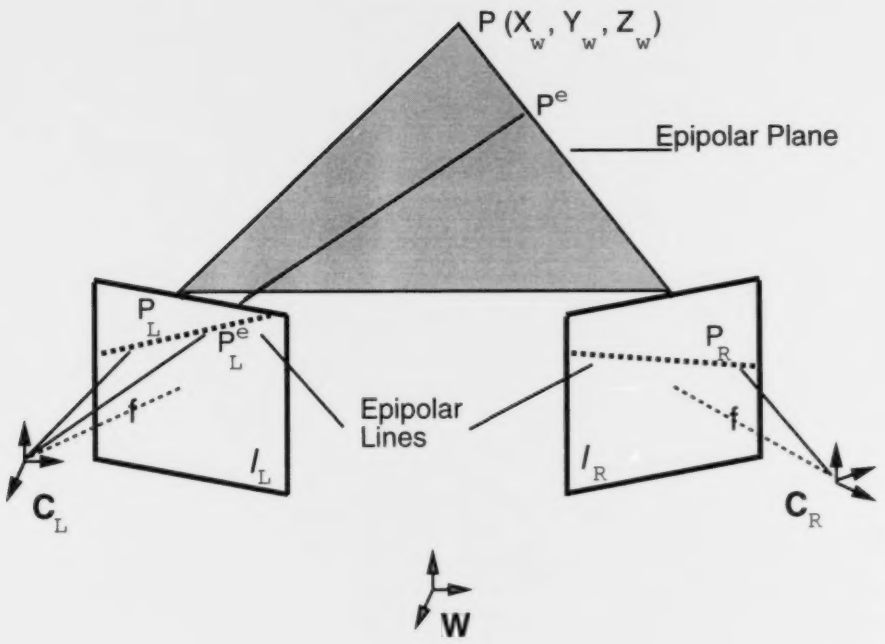

Figure 3.1: *Perspective Projection with Stereo Geometry.*

projected image point.

- Make adjustments for lens distortions, sensor discretization, principal point (real image center), atmospheric refraction, *etc.* These are specific to the camera equipment being used and the modelling of this equipment based on the accuracy required.

Most camera calibration methods calculate camera parameters based on such a mapping. Their models differ due to the differences in the adjustments that they may make. Some terminology related to this process is given here:

Coordinate Systems In a stereo-system, 3D points can be described in any one of three coordinate systems - world (object-space), left-camera, or right-camera. The origins of these systems are W, C_L, and C_R in figure 3.1. Transformations between these coordinate systems are possible if the extrinsic parameters of both cameras are known.

Transformation/Perspective Matrix The matrix which maps 3D object-space points to 2D image points. The exact form will depend on model assumptions.

Extrinsic Parameters The 6 parameters which describe the position (X,Y,Z) and orientation (ω, ϕ, κ) of a camera.

Intrinsic Parameters Camera parameters depending on the analytical model used to relate 3D object-space and the camera's 2D image plane. They vary in number depending on the assumptions made in the model, usually including factors for radial lens distortions (K_1, K_2), decentering lens distortions (P_1, P_2), principal point location (x_0,y_0), and focal length (f).

Epipolar For an object-space point, P (fig. 3.1), the *epipolar plane* is the plane containing the points P-C_L-C_R. *Epipolar lines* are the intersection of this plane with the two image planes in a stereo-system. The irreversibility of the 3D-2D mapping for a single camera is also evident from the figure as the single image point P_L can be a projection of the infinite number of object-space points lying on the line PP_L.

Baseline The distance between the camera perspective points, C_L and C_R, is termed the baseline of a stereo-system. Larger baselines result in improved triangulation (reconstruction) accuracies, but lead to perspective views which differ greatly. This impedes automatic correspondence between image points in the two views.

During calibration, when dealing with cameras and lenses, there were some important factors which came into play.

Focal Length The perpendicular distance of the perspective center (C_L) from the image plane I_L (fig. 3.1).

Range The distance between cameras and the object being imaged.

Field of View (FOV) The dimensions of the rectangular plane under focus at the prescribed range.

Depth of Field (DOF) If a lens is focused at a particular range and targets at that range are shifted either away or toward the camera, the corresponding image points become blurred. Depth of field, for a given focal setting, is the deviation from the ideal range which result in acceptable defocussing of image points. DOF inversely depends on aperture and focal length [7].

Depth of Focus Depth of focus refers to the distance the image plane can be moved for a fixed object and still provide for acceptable focussing. This is opposed to depth of field which depth of field refers to the distance the object can move with the image plane fixed so that the resultant defocussing is acceptable.

Aperture Stop (f/d) The aperture stop is given by the ratio between the focal length and aperture diameter of the camera. Standard stops consecutively increase by a factor of $\sqrt{2}$ (1, 1.4, 2, 2.8, 4, 5.6, 8, 11, 16 ..). This implies that the aperture diameter reduces by a factor of $\frac{1}{\sqrt{2}}$ between stops and light, proportional to the square of the aperture diameter, is reduced by a factor of $\frac{1}{2}$ with each stop. The aperture setting proves crucial for setting the DOF for different lighting considerations.

Experimentation in the OR led to a set of optimal camera and lens settings (table 3.1). Machine vision software was developed using images taken by the system in these settings.

Methods

Intraoperative monitoring was to be done using off-the-shelf CCD cameras with limited resolution. These cameras are classified as non-metric cameras [59] and require sophisticated modelling for calibration. A survey of camera calibration methods [9] was conducted to arrive at a suitable calibration approach. Three methods showed promise. Tsai [36, 62] presented a geometrical model for calibration called the *two-stage* method. It was based on the radial alignment constraint. The popularity of this method in computer vision stems from the fact that unlike photogrammetric approaches which deal with plate coordinates, Tsai considered the finite spacing between sensor elements in CCD cameras and digitizers. The approach of Peterson *et. al.* [51]

used the classical collinearity principles to determine camera parameters [70]. An interesting approach by Wei and Ma [65, 66] involved the implicit determination of the line-of-sight (LOS) for any image point. This approach used interpolation methods with calibration points on two-planes to determine the LOS vector. Thus, no camera modelling was required. The drawback of this approach is the larger computational and memory expense relative to the other two methods.

As his software was available, Peterson's approach [51] was tested. It yielded accuracies better than 1 mm at the specified range of 700 mm; accuracies comparable to those reported by Tsai's method. The two-plane methods were not feasible because of their computation and memory loads during 3D point reconstruction.

3.2 Camera Calibration

The approach to calibration used by Peterson *et. al* [51] was used to model the stereo-system in the OR. This approach first uses the Direct Linear Transformation (DLT) [1] to obtain estimates of camera parameters. These estimates are then fed as approximations to linearized collinearity equations which yield the final parameters on iteration. It is necessary to use collinearity to refine the parameters returned by the DLT because it has been reported [62] that the accuracy of reconstruction, when using parameters returned by the DLT, fell by over 100% outside the central (90%) FOV. This section provides the theoretical background to the DLT and collinearity principles. A simultaneous least squares adjustment of all the measurements in a photogrammetric mapping problem can be formulated by the use of condition and observation equations [59]. This section only provides the mathematical model of each system of equations. The complete treatment of these methods, in terms of formation of observation equations, linearization, least squares adjustment and error analysis, is beyond the scope of this work. Simulations of object-space point projections and image point reconstructions were widely used during the development of the OR-system. The basis for these simulations are also provided.

3.2.1 DLT

The Direct Linear Transformation, or DLT, is a system of equations which represents a direct transformation between image coordinates and object-space coordinates without explicitly considering interior orientation parameters [1]. The later form of the DLT [59] includes constraints of interior orientation and is no longer a linear system though it has maintained the name. The model of the DLT used herein accounts for intrinsic parameters and has the following form:

$$\begin{aligned} x + \delta_x &= \frac{L_1X + L_2Y + L_3Z + L_4}{L_9X + L_{10}Y + L_{11}Z + 1} \\ y + \delta_y &= \frac{L_5X + L_6Y + L_7Z + L_8}{L_9X + L_{10}Y + L_{11}Z + 1} \end{aligned} \tag{3.1}$$

where, $\delta_x = (x - x_0)K_1r^2, \delta_y = (y - y_0)K_1r^2$; (x,y) are the measured image coordinates; (x_0, y_0) is the principal point, or image center; (X, Y, Z) are the corresponding object-space coordinates; $r^2 = x^2 + y^2$, the radial plate distance; K_1 is the radial lens distortion coefficient (other radial coefficients, $K_2, K_3, ..$, and decentering coefficients, $P_1, P_2, P_3, ..$, are usually too small to have a significant distortion effect [51, 70]); $[L_1L_2L_3 \ldots L_{11}]$ are eleven unknown constants. As each image point yields 2 equations using equation 3.1, a minimum of 6 *well distributed* control points are required to solve for the 12 unknowns, $L_1, ...L_{11}, K_1$. All extrinsic and intrinsic parameters can be computed from these solutions [46]. The system of equations for n ($n > 6$) control points will have redundancy and can be solved directly or iteratively [59, pg. 801].

A FORTRAN implementation of the DLT (NDLT) [50] was used to provide parameters estimates used as approximations with the collinearity conditions.

3.2.2 Collinearity Equations

The collinearity condition requires that the perspective center of the camera, image point (positive image convention was used), and the object point in three dimensional reference space all lie on the same straight line (fig. 3.2). In the camera coordinate system, centered at the perspective center(O_C), image points, (x,y), can be represented by the vector, $\mathbf{V} = <$x y -f$>^T$, where f is the principal distance of the camera.

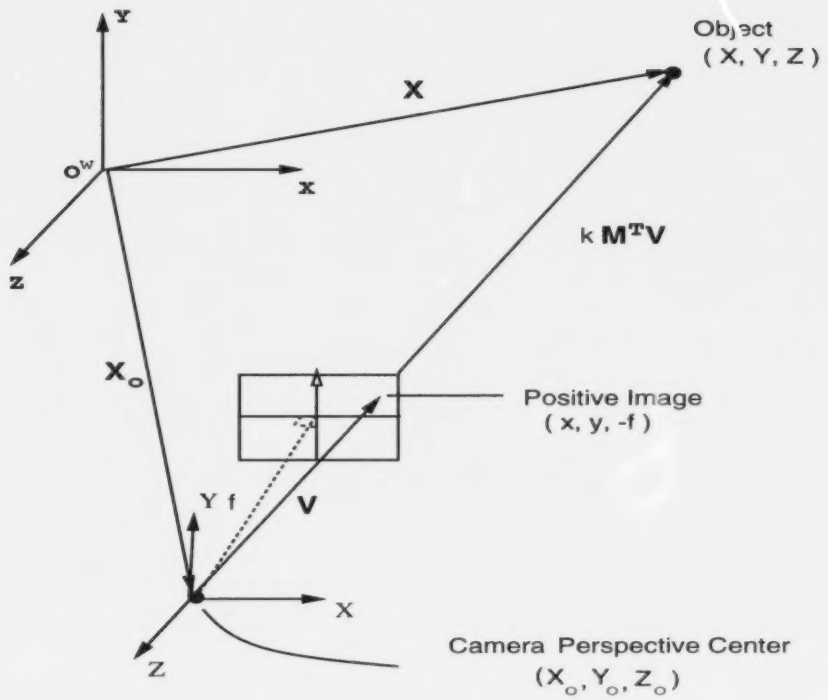

Figure 3.2: *The Collinearity Condition : Convention and Terminology.*

This is the same as the *calibrated* focal length if the camera has a fixed focus lens. In the object-space coordinate system (center O^W), the perspective center can be referred to by the vector, $\mathbf{X_0} = (X_0, Y_0, Z_0)$, and the object point is represented by $\mathbf{X} = (X, Y, Z)$. By the collinearity condition, the vector $(\mathbf{X} - \mathbf{X_0})$ is collinear to the vector $(\mathbf{M^T V})$, the vector $\mathbf{V}$ rotated into the object-space coordinate system by the inverse of the orthogonal matrix, $\mathbf{M}$. Collinear vectors, in the same system of reference, only differ by a scalar factor. Hence,

$$(\mathbf{X} - \mathbf{X_0}) = k\mathbf{M^T V} \tag{3.2}$$

where $\mathbf{M} = \mathbf{R_z R_y R_x}$ is the product of the three orthogonal matrices $(\mathbf{R_z}, \mathbf{R_y}, \mathbf{R_x})$ corresponding to successive rotations (ω, ϕ, κ) about the x, y, and z axes, in that order. Rearranging and expanding the matrices in eq. 3.2, we have the collinearity equations as:

$$\begin{bmatrix} x \\ y \\ -c \end{bmatrix} = (1/k) \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix} \begin{bmatrix} (X - X_0) \\ (Y - Y_0) \\ (Z - Z_0) \end{bmatrix} \tag{3.3}$$

Eq. 3.3, a system of three equations, is difficult to use because the scalar factor (k) is different for each object-space point. To eliminate this factor, eq. 3.3 is reduced to a system of two equations by dividing the first and second equations by the third. Also, if the equations are revised to consider the principal point (actual image center) and second-order radial lens distortions, the two collinearity equations can be stated as:

$$\begin{aligned} x - x_0 + \delta_x &= (-f) \left[\frac{m_{11}(X - X_0) + m_{12}(Y - Y_0) + m_{13}(Z - Z_0)}{m_{31}(X - X_0) + m_{32}(Y - Y_0) + m_{33}(Z - Z_0)} \right] \\ y - y_0 + \delta_y &= (-f) \left[\frac{m_{21}(X - X_0) + m_{22}(Y - Y_0) + m_{23}(Z - Z_0)}{m_{31}(X - X_0) + m_{32}(Y - Y_0) + m_{33}(Z - Z_0)} \right] \end{aligned} \tag{3.4}$$

where, principal point, (x_0, y_0), and radial distortion components, (δ_x, δ_y), are the same as in eq. 3.1.

A FORTRAN implementation of the collinearity equations (V2STEREO) [50] was used for this work. V2STEREO uses a Taylor series linearized form of the collinearity equations (eq. 3.4). Results are presented at the end of this chapter.

3.3 Results

This section provides photogrammetry-related data about the stereo-vision system used in the operating room. It describes the equipment used, camera geometry suitable for intraoperative monitoring, the calibration procedure, and simulations done. Calibration results are also provided.

Equipment Used

The system (fig. 2.6) consisted of a pair of Sanyo VDC-2524 cameras, Tamron Variable Focal Length (6-16mm) lenses, a Scion II Image Capture Card used with a Macintosh computer to capture images intraoperatively, and an Optex VA0150 Video Halogen Lamp (table 3.1). Research work and software development was done separately on an IBM RISC/6000. All software, except V2STEREO and NDLT, was developed in-house using the ANSI C programming language. Table 3.1 lists the camera geometry

SYSTEM EQUIPMENT	
Cameras	Sanyo VDC-2524, B/W, NTSC video output
Camera Sensor	CCD frame transfer, 800x500 pix., 6.4x4.8 mm
Lenses	Tamron (f=6-16mm), variable aperture
Lamp	VA0150 Optex 150W/120V Halogen Video Lamp
Frame Grabber	Scion II Image Capture Card (640x480 pix.)
Captured Image	B/W, 568x416 pix. (subset of Frame Grabber FOV of 640x480 pix.)
GENERAL CAMERA GEOMETRY	
Range	630-760 mm
Focal Length	7-7.5 mm
Field of View (FOV)	500x400 mm
Depth of Field (DOFi)	±75 mm
Aperture	midway (varies)
Baseline	300-500 mm (varies)
Vergence	$15 - 20°$

Table 3.1: *System Equipment and Camera Geometry.*

experimentally found to be optimal in the OR. This geometry proves useful when the system is to be recalibrated or changed.

Simulations for Control Field

A control frame which was *geometrically suitable* for calibration of the camera geometry chosen for the operating room (table 3.1) was designed and constructed. This *suitability* was tested by simulations. A control frame was simulated as a grid of 30 points, 5 rows of 6 points, with widely varying heights from 0 mm to 105 mm. The heights roughly corresponded to a LOW-HIGH pattern where, if one point was to be LOW, all surrounding points were to be HIGH and *vice versa*. A LOW height was a choice from the set {0, 15, 30, 45} mm and a HIGH height consisted of one of {60, 75, 90, 105} mm. The simulations used cameras conforming to the general camera geometry and the collinearity equations (eq. 3.4). It was found that peripheral points of the control frame should have LOW heights to be easily accommodated in calibration stereo-images.

Table 3.2 lists the simulated configuration of 30 points which was found to be stable, i.e. calibration runs easily converged when using simulated projections, with random error, of the control field. It also lists the actual point coordinates after

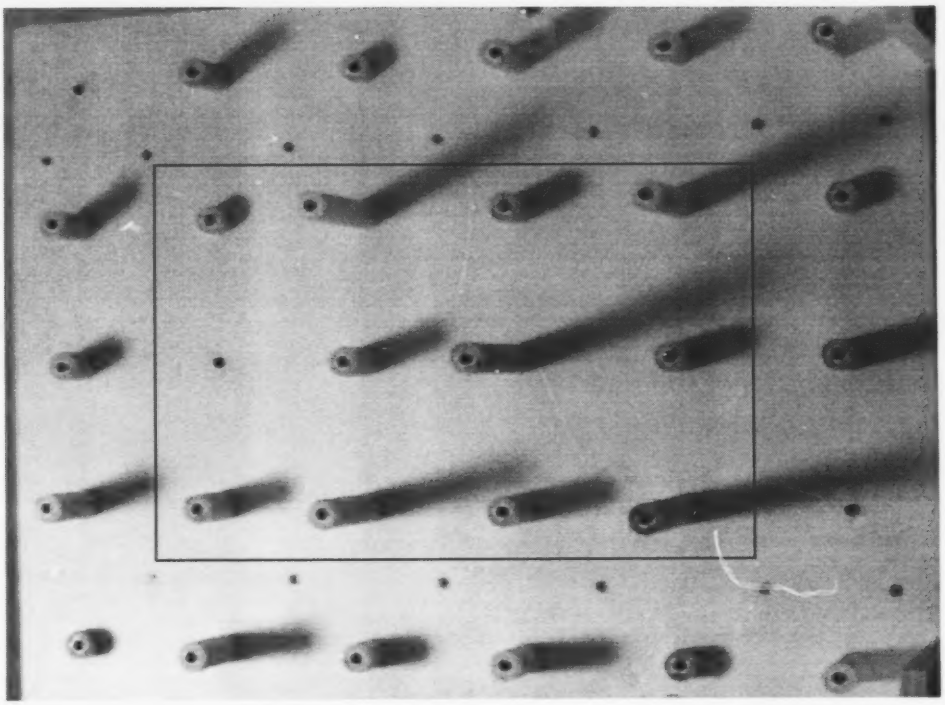

Figure 3.3: *The* Peg Board *with 30 points (tab. 3.2). A typical 3x4 subset used for calibration is indicated.*

construction of the control grid as a 'peg board' (fig. 3.3) with 'pegs' of different heights holding different points.

Calibration Procedure

The calibration procedure consisted of taking images of a subset of 12 points of the control field (*peg board*) and using calibration routines to arrive at specific camera parameters. Experience with calibration has shown that with the OR camera geometry, 3 rows of 4 points sufficiently covers the entire FOV (fig. 3.3). Multiple shots were taken for calibration verification.

Including accuracy assessment, there were five stages for calibration. Along with the routines they use, these can be listed as:

1. Image Acquisition and Digitization of control points,

Control Point	$X_s = X_c$	$Y_s = Y_c$	Z_s	Z_c
1	0	508	0	0.000
2	127	508	45	44.780
3	254	508	15	14.858
4	381	508	45	44.754
5	508	508	30	29.794
6	635	508	60	59.893
7	0	381	45	44.780
8	127	381	15	14.884
9	254	381	105	104.902
10	381	381	30	29.921
11	508	381	90	89.839
12	635	381	15	14.732
13	0	254	30	29.895
14	127	254	0	0.000
15	254	254	45	45.110
16	381	254	120	120.192
17	508	254	30	30.073
18	635	254	45	44.856
19	0	127	60	59.994
20	127	127	45	45.059
21	254	127	90	90.195
22	381	127	45	44.983
23	508	127	105	104.952
24	635	127	0	0.000
25	0	0	15	15.113
26	127	0	60	60.248
27	254	0	30	30.124
28	381	0	45	44.932
29	508	0	15	14.985
30	635	0	60	59.918

Table 3.2: *Control Field : Simulation Coordinates* (X_s, Y_s, Z_s) *and Constructed* (X_c, Y_c, Z_c) *(*Peg Board*) Coordinates. Only the constructed depths differed from the simulation depth specifications (All coordinates are in mm).*

2. **PIX2MM** C routine used to convert xv[1]-based digitized control point coordinates from pixels to mm on the sensor,

3. **NDLT** FORTRAN (DLT) routine used with plate coordinates to arrive at initial approximations of parameters,

4. **V2STEREO** FORTRAN (collinearity) routine used with plate coordinates and initial approximations to arrive at final parameters,

5. **SIM3D** FORTRAN routine used with final camera parameters to transform a set of derived coordinates and compare to control point coordinates to evaluate accuracy of calibration.

The program PIX2MM.c was developed for the Sanyo camera image sensor. This program converts image pixels to millimetres on the sensor; there is also an important adjustment for scale which corrects for the aspect ratio. For the Sanyo cameras, this factor was experimentally determined to be 0.845. The factor was determined as the average ratio of the principal distance in the x-direction to that in the y-direction (eq. 3.5) for a number of NDLT runs.
For Sanyo VDC-2524 CCDs,

$$scale_factor_Y = \frac{f_x}{f_y} = 0.845 \tag{3.5}$$

The other function of PIX2MM was to transform points from pixels in the xv image-origin (in the top-left of the image) to millimetres in the plate-origin in the center (with Y-axis inverted). Theoretically, this conversion to plate coordinates seems unnecessary, but it was found that when pixels were used directly, a different scale_factor_Y was required and V2STEREO often did not converge.

Five calibration trials (A, B, C, D, and E) were conducted with different control frame images. All trials converged. Based on these trials, the camera parameters

[1]xv is the X-windows program used used for viewing and manipulation of images of different formats (RAS, TIF, GIF, *etc.*). The origin for xv's pixel coordinate system is at the upper left corner of the image. The camera 'plate' has its origin at the center of the plate so that the top right corner of the image is in the first quadrant.

EXTRINSIC PARAMETERS (ω, ϕ, κ in DDDMMSS.X)							
CAMERA	X (mm)	Y (mm)	Z (mm)	ω	ϕ	κ	
LEFT	223.9545	442.5372	743.9232	-50609.0	-165732.2	12336.1	
RIGHT	645.7806	442.3287	734.5083	-42648.9	151302.0	-15556.3	
INTRINSIC PARAMETERS							
CAMERA	f (mm)	$x_0(mm)$	$y_0(mm)$	$K_1(mm^2)$	$K_2(mm^2)$	$P_1(mm^2)$	$P_2(mm^2)$
LEFT	7.776	.061	.078	.29E-02	.42E-26	.53E-26	.16E-25
RIGHT	7.556	.018	.090	.33E-02	.55E-25	.60E-25	.21E-25

Table 3.3: *Calibration Parameters for trial C.*

RECONSTRUCTION ACCURACY (using Trial C parameters, table 3.3)			
TRIAL	X (mm)	Y (mm)	Z (mm)
A	0.3506	0.5456	0.8258
B	0.3895	0.2761	0.7108
C	0.2351	0.3060	0.4807
D	0.3473	0.3297	0.7106
E	0.3666	0.2683	0.5696
AVG.	0.3378	0.3451	0.6595

Table 3.4: *Reconstruction Accuracy using trial C parameters. The standard deviation of residuals is quoted for the reconstruction of each trial. The mean of residues was zero in each case.*

for trial C and accuracy of reconstruction of control points in all calibration runs are given in table 3.4. The mean error of reconstruction in each case was free from bias (mean residual on x, y, and z was 0.0000 mm [50]). The quoted accuracies are the standard deviations on x, y, and z residuals as calculated using the FORTRAN program, SIM3D [50].

During the design phase of this project, concerns were raised [21] regarding the accuracies of measurements possible using 'surveillance-type' off the shelf cameras and lenses. Successful calibration experiments and verification done have proved that the photogrammetric techniques used [50] are capable of providing submillimetre 3D measurements in the range of 500-900 mm with baselines as wide as 500 mm. With rudimentary knowledge of the photogrammetric principles and software, it is possible to calibrate the cameras with ease. It has been suggested as part of future work that the different calibration programs be integrated and an automatic 'shoot and calibrate' program be developed. Future calibration models should also account for the discretization of the image sensor in CCD cameras [62].

This chapter dealt with the use of stereo-cameras to make 3D measurements. Testing involved the manual digitization of control and target points in images. In the operating room, manual digitization of features in stereo-images is inconvenient. Work was done towards the automatic detection of the spherical-components of triad markers (called ball-features) in OR images. The next chapter focuses on experiments and algorithms for automatic feature detection.

4 Feature Recognition and Low Level Processing

Vision systems first attempt to use low level image processing techniques, those dealing with pixels and edges, to extract information from a scene [64]. This information is represented in structures called features which are then used by higher level processes to make more advanced inferences about the scene. The primary objective of the system developed in this study is to reconstruct triad-markers (fig. 4.1) in 3D. To obtain information about a triad and its location in three dimensions, we need to locate its spherical components (ball-features) in both images. Thus, provided there is no occlusion, feature recognition should locate at least 3 ball-features in each image corresponding to every triad placed on the spine. Higher level matching processes can then proceed with reconstruction in 3D based on the correspondence between ball-features in the left and right image.

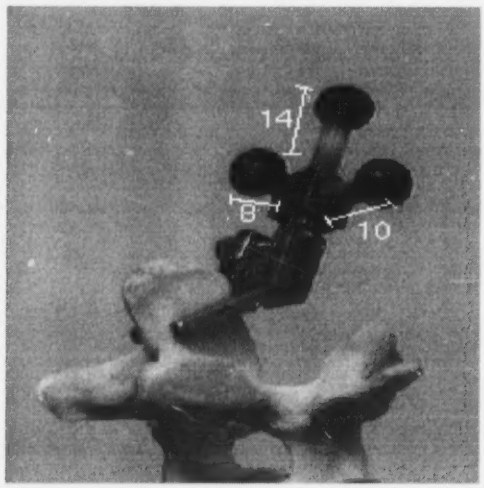

Figure 4.1: *A triad-marker model. An orthogonal configuration of spherical balls is shown here. Configurations with spherical balls in a flat, planar configuration were also used. The spherical components are 8mm in diameter (all model dimensions are in mm).*

This chapter focuses on algorithms and experiments used to develop a procedure for the detection of ball-features in operating room (OR) images. The software developed for this purpose was called DETECT-POINT. This first section describes the analysis of typical operating room images. The discussion is based on exist-

ing methods for feature recognition. A model-based approach, implemented using template matching, was adopted. Subsequent sections deal with the algorithms and experiments used to develop DETECT-POINT. Preprocessing of raw images from the OR is presented in detail. This consists of various transformations like gamma correction, image size reduction, mode filtering and edge detection. These operations modify raw images, making them more suitable for the feature detection procedures to follow. Details of the design and implementation of the template matching, the chosen approach, are then presented. Implementation details are explained using comparisons with alternative methods and experimental results on OR images. A mathematical formulation is presented which applies cross-correlation in the spatial frequency domain. This could be applied to improve the speed of DETECT-POINT which currently relies on a spatial domain implementation of correlation which is computationally expensive.

DETECT-POINT was tested on images from a three-stage surgery. The first stage was used to design models used by DETECT-POINT; the program was then applied to all three stages. Results of this experiment are provided.

4.1 Analysis of Operating Room Images

A typical stereo-pair of operating room images, after contrast adjustment, is shown in figures 4.2 and 4.3. This section presents an analysis of such images and their properties to discuss the options available for ball-feature detection. All references to images, features, background *etc.* implicitly refer to the above mentioned figures.

The histogram of these images (fig. 4.4) reveals that features, which appear as circular, specular gray regions (fig. 4.2) have intensity levels in common with the background, but occupy a small area of the image. Consequently, they do not exclusively occupy any mode in the histogram. This rules out the possibility of using any global histogram or threshold selection methods [26] to segment these images into background and foreground (features). Local methods [68] based on the same principles will fail due to insufficient contrast between the features and their backgrounds. Local histograms, taken over smaller subimages, have little informative modal distri-

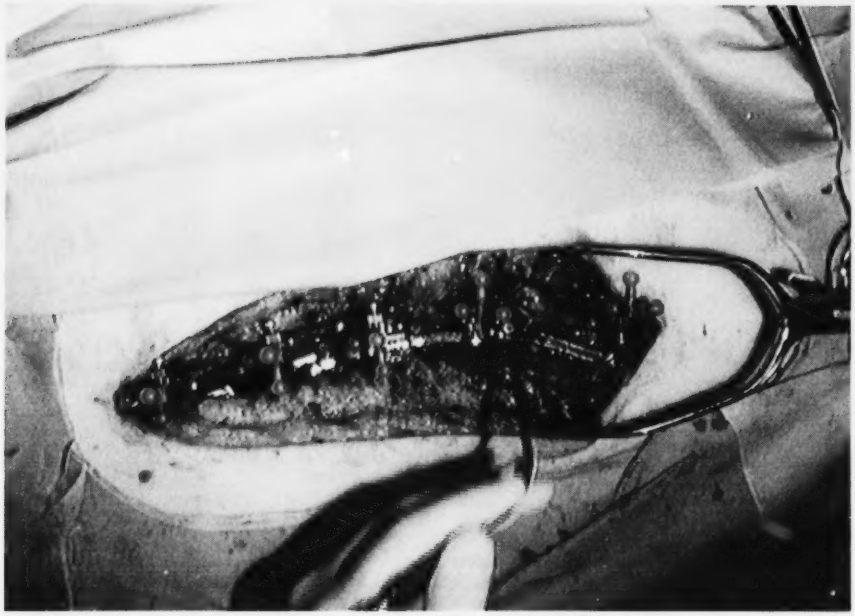

Figure 4.2: *Contrast Adjusted Operating Room Image (Left Camera).*

Figure 4.3: *Contrast Adjusted Operating Room Image (Right Camera).*

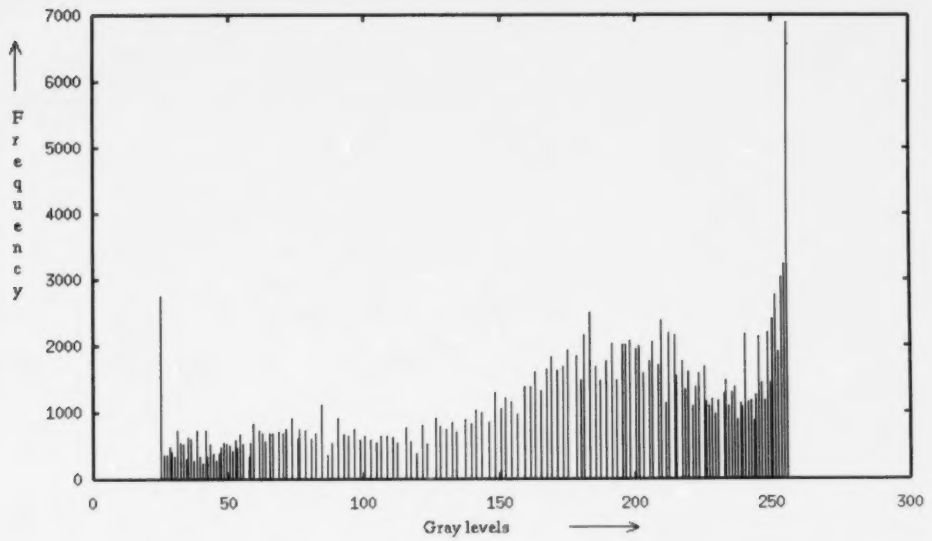

Figure 4.4: *Histogram of Preprocessed Right OR Image. Feature gray levels are embedded in the range 100-230, however, they account for only a small percentile of the total number of pixels in that intensity range.*

bution even though feature pixels account for most of the histogram. As histogram or region-based segmentation is not robust enough, shape-based methods which proved successful in an earlier pilot study [8] were ruled out. The pilot study used a spine model with high contrast markers. Size and moment constraints were applied on connected components of binary images to identify possible features. Region growing methods [4] are also not feasible due to the difficulty of automatically assigning seed locations when ball-features occupy an unpredictably small percentage of image area.

Edge detection and the Hough transform [18] showed promise but failed to yield consistent results. This was because the small size of the feature boundary (diameter 10 pixels or 8 mm in the model) and noisy surgical field of view yield a 'highly excited' Hough-parameter space. Segmentation of this space provides little relevant information.

Photometric methods [23] would require the modelling of the light source, precise knowledge of the location of cameras and the reflectance properties of the surgical field of view. As the background of flesh, blood, bone and instrumentation cannot be

characterized easily in this way, this approach is also not feasible.

Texture and entropy methods [69] show promise if the fleshy background can be characterized statistically with texture features different from those of the more consistent balls features. However, research in this area is still not considered mature [37, 69] enough to apply successfully to complex environments like the operating room.

A model-based approach, using multiple templates and image matching techniques, seemed more appropriate than others. This is because many characteristics of the ball-features such as size, shape, specular appearance, and, in most cases, significant contrast between the feature, and its background are suited to template matching [13, 60].

The following sections elaborate on the rationale behind the different components of our approach as implemented in the DETECT-POINT software. These are supported by experimental results on OR images.

4.2 Preprocessing

Preprocessing of raw images in DETECT-POINT consists of a number of operations, each with its own objective. The first operation applied to these images is gamma correction. Gamma correction improves contrast levels of raw images. As OR images capture more than the surgical field of view, a routine was implemented to automatically identify the region of interest. This results in reduced processing time. Median filtering improves the signal to noise ratio of the images by removing 'impulse' noise. The filter implemented for this purpose implicitly sharpens edges also. Edge images are required by later stages of detection. These are also computed as part of preprocessing.

4.2.1 Gamma Correction

In the OR, it is important to keep the aperture of the cameras small for greater depth of field. However, images taken with small camera apertures are dark (fig. 4.5). Such low contrast images do not use the entire intensity range of the sensor. The variac-

controlled lamp, positioned between the cameras (fig. 2.6), is designed to offset this drawback. By increasing the intensity of the lamp, brighter images can be obtained while keeping the aperture small. This condition, however, does not apply beyond the saturation of the sensor due to glare. Iterated gamma correction (eq. 4.1) was used to improve the contrast levels of raw OR images.

After the i*th* iteration, the output intensity v_i is given by:

$$v_i = 255 * \left[\frac{v_{i-1}}{255}\right]^{1/\gamma} \tag{4.1}$$

where γ is the gamma correction index and, for the first iteration (v_1), v_0 (on RHS) is defined as the raw image intensity.

Historically, *gamma* refers to the quality of contrast produced by a photogrammetric film. High-contrast films reproduce tone differences in the subject as large density differences in the photograph, while low-contrast films translate tone differences as small density differences. Characteristic curves for films plot density against log of the exposure (D-logE). These curves, called H & D curves [64] (after Hurter and Driffield, who developed this method) correspond to response curves of modern sensors. CCD sensors have a *gamma*, measured using equation 4.2, approximately equal to 1.0, while vidicon sensors have *gamma* values close to 0.65. Eq. 4.1 represents a special form of the generalized gamma expression (eq. 4.2).

$$\gamma = \frac{log(\text{sensor output signal})}{log(\text{sensor input signal})} \tag{4.2}$$

Gamma correction was applied to the raw image in figure 4.5. Figures 4.6 and 4.7 show results after the first and second iterations respectively. Gamma indices of 2.0 and 1.5 were used for iterations one and two respectively. It was also found that two iterations proved more effective than a single iteration with a high index.

4.2.2 Automatic Size Reduction

Only about a third of a typical OR image consists of the surgical field of view (FOV). As seen in figure 4.5, the image is well approximated by the length of the surgical FOV. However, width-wise, most of the image consists of irrelevant portions of the

Figure 4.5: *Raw Operating Room Image.*

Figure 4.6: *Image after Gamma Correction, iteration #1 ($\gamma = 2.0$).*

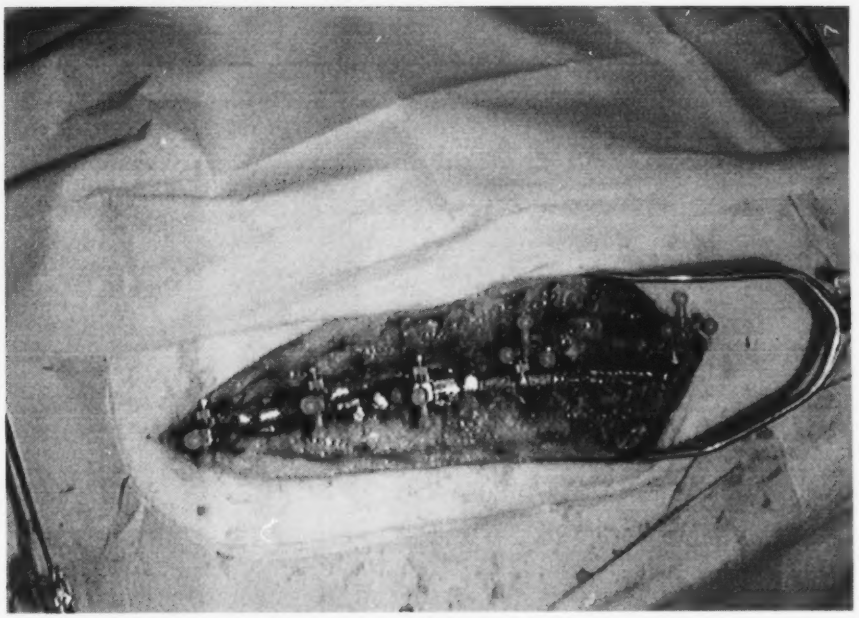

Figure 4.7: *Image after Gamma Correction, iteration #2 (γ= 1.5).*

patient's back. It would be unwise to subject such images to computationally expensive procedures like template matching. Thus, a method was developed for the automatic identification of the region of interest, or surgical field of view, in operating room images.

The method considers basic statistical parameters of image segments. Image segments are defined as collections of 5 contiguous rows, each spanning the length of the image. Hence, our images of size 568x416 pixels have 83 segments of size 568x5 pixels and one boundary segment of size 1x568 pixels. Averages and standard deviations are calculated for each segment of the image. The distributions for averages and standard deviations of the segments of figure 4.7 are shown in figure 4.8 and figure 4.10 respectively. Thresholds are applied to these distributions to obtain isolated modes. These isolated modes correspond to groups of image segments which survive thresholding of the distribution. The surgical field of view was considered to be the region of the image with the highest standard deviation and lowest average. Hence, if we consider the distribution of standard deviations of image segments, the largest central mode in the distribution would be the *mode of interest.* This could be taken to

directly correspond to the surgical FOV. On the other hand, if we consider averages, the largest central trough in the distribution of image segment averages would qualify as the *mode of interest.*

Regions of low averages could also develop if the surgical drape covering the patient becomes stained. The standard deviation was thus chosen as a more robust parameter than the average. This is seen in figures 4.8, 4.10. Threshold selection for the distributions is dynamic and is a function of the maximum and minimum parameter values (eq. 4.3).

Given the distribution of a parameter (average/standard deviation), the threshold value, T, is chosen as:

$$T = max - \alpha(max - min) \tag{4.3}$$

where max, min are the maximum and minimum parameter values of the distribution and α is the fraction of the range to be considered. An α of 0.5 produced good results (figs. 4.9, 4.11).

On thresholding, if the distributions are found to contain multiple modes (fig. 4.9) the *mode of interest* is selected as the mode containing the maximum standard deviation. The subimage containing the rows corresponding to this selected mode of image segments is the automatically detected surgical field of view. Figures 4.12 and 4.13 show an original image and its automatically detected region of interest respectively.

4.2.3 Mode Filtering

Gamma correction improves the contrast of images considerably. However, it also degrades the signal to noise ratio of already noisy images. In this context, noise refers to unnecessary gray level variations within a small spatial area. This variability in images makes them unsuitable for the task of model-based recognition. Hence, preprocessing must include operations to reduce the noise level of the image.

Smoothing is a common technique used to improve noisy images. The most common smoothing operations in the spatial domain are implemented using the mean and median filters [29]. However, if preprocessed images are to be used to make precise

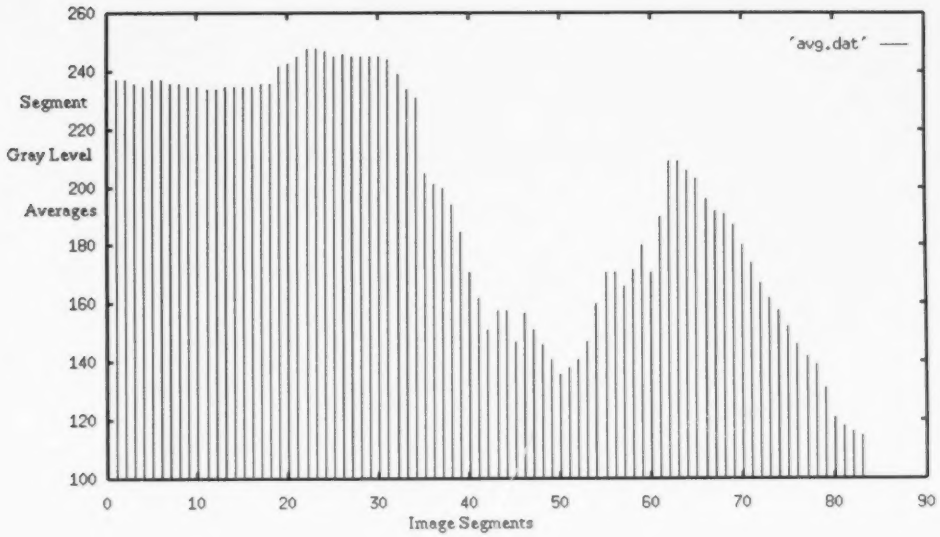

Figure 4.8: *Distribution of Averages of Image Segments for fig. 4.7.*

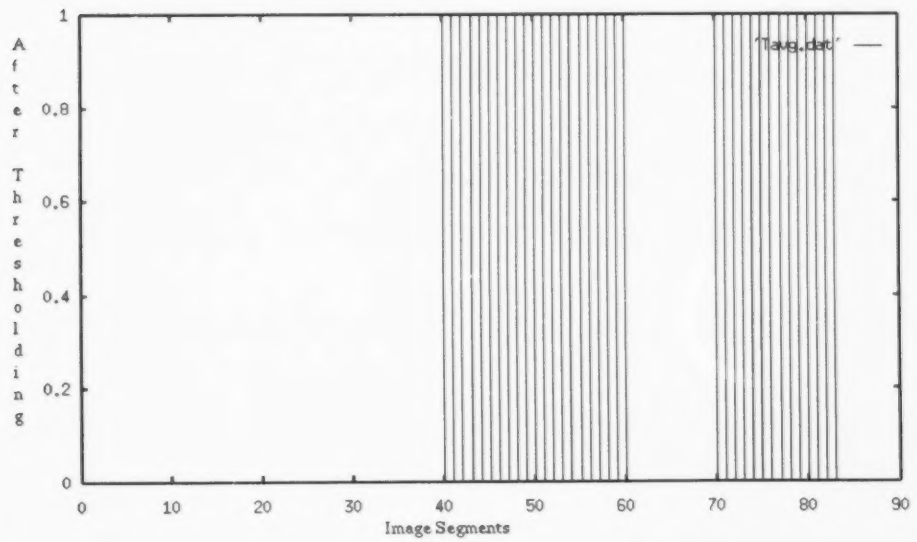

Figure 4.9: *Averages distribution (fig. 4.8) after thresholding. The threshold was calculated to be 185 using* $\alpha = 0.5$.

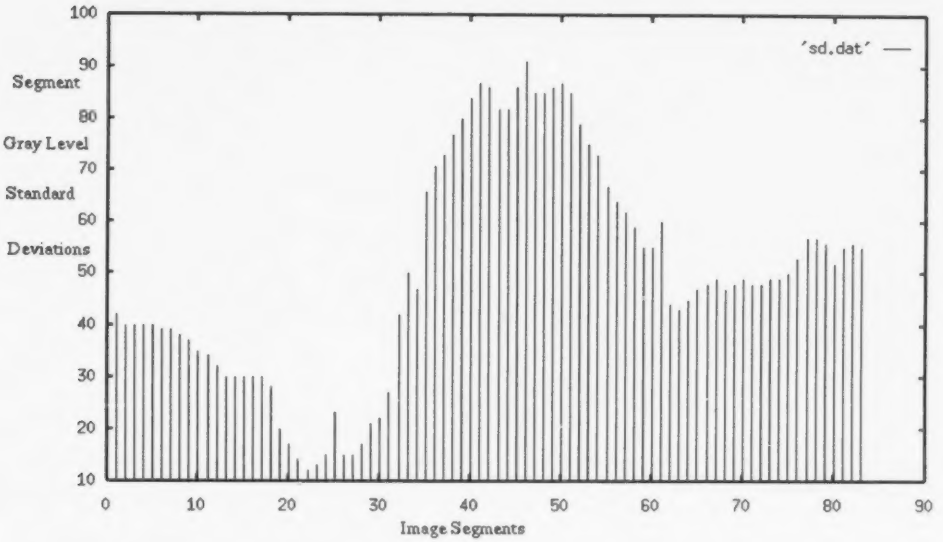

Figure 4.10: *Distribution of Standard Deviations of Image segments for fig. 4.7.*

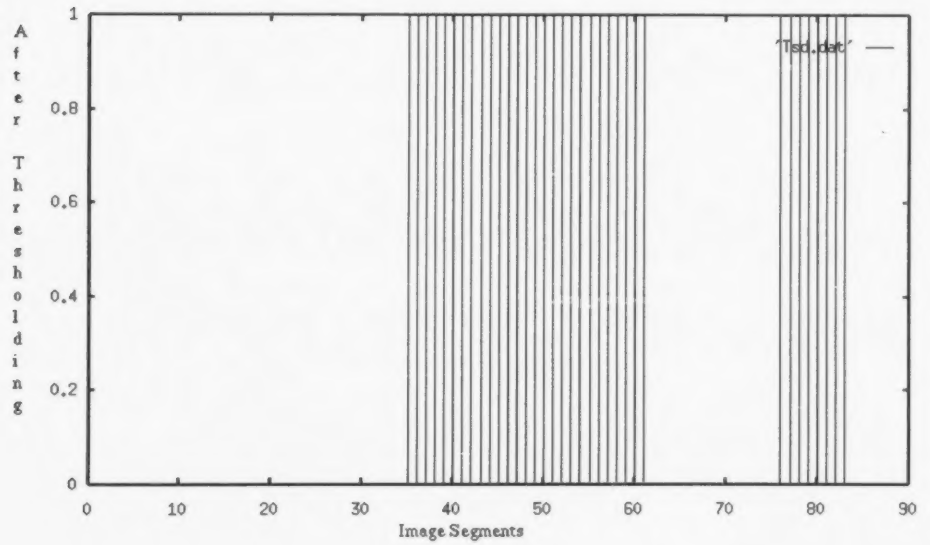

Figure 4.11: *Standard deviation distribution (fig. 4.10) after thresholding. The threshold was calculated to be 53 using* $\alpha = 0.5$.

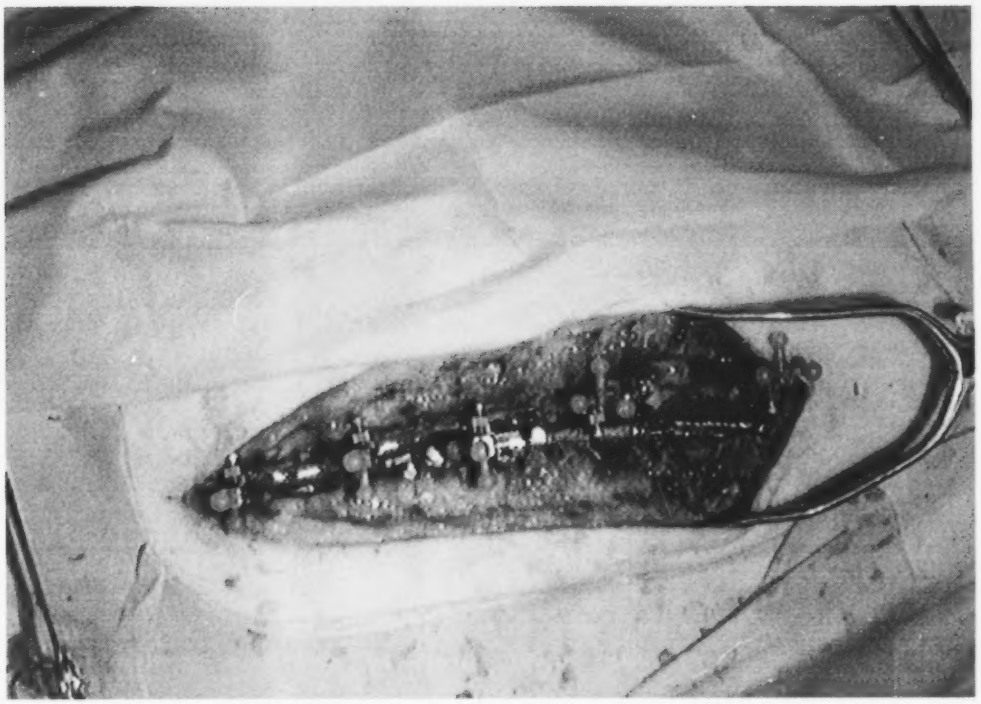

Figure 4.12: *Original OR Image.*

Figure 4.13: *Automatically detected surgical field of view for fig. 4.12.*

measurements, we must first assess the effects of different smoothing operations on feature positions.

Mean or Gaussian filtering, which is equivalent to low pass filtering in the frequency domain, reduces the high definition content of images and produces blurred images. This is undesirable as blurred images have low edge content and subsequent correlation methods (pg. 52) rely heavily on the magnitude and position of edges to infer the position of features. Any smoothing operation applied at this stage must not effect either of the edge properties (viz. position, magnitude) or the underlying gray value distribution of a region. Another disadvantage of these filters is that impulse noise, which is produced by the sensor and needs to be removed completely, is allowed to 'smear' over local neighbourhoods [28]. Median filtering is an effective way of suppressing such impulse noise. The images produced by median filters have a 'softened' appearance and have little loss of detail.

Recently, Davies [17, 18] introduced the mode filter as a viable alternative for noise smoothing. The mode represents the most probable value of any distribution, a more meaningful parameter than the mean or the median. However, its implementation for small neighbourhoods like 3x3 pixels presents a tedious problem when the are 256 possible levels for each of the 9 pixels in that neighbourhood. Unlike a continuous distribution where the mode is easily locatable, the intensity with the highest frequency in such small, discrete distributions (fig. 4.14) is not necessarily the *underlying* mode. This is equivalent to a continuous distribution with a superimposed impulse (fig. 4.15) higher than the real mode of the underlying distribution. A width of 256 gray levels for a 3x3 distribution is another anomaly. For these reasons, an indirect estimate of the mode is sought rather than a direct calculation [16]. The mode filter was implemented by calculating the median as an estimator of the mode and removing a portion from a end of the distribution. This truncation was done (at T_a) so that the ends of the resultant distribution are equidistant (L) from the median of its parent distribution (fig. 4.16) . The median of this new, truncated distribution is now closer to the mode than its predecessor. Iteration, in this fashion, leads to a practical estimate of the mode.

Figure 4.14: *The sparse local intensity histogram of a 3x3 neighbourhood where the orthodox mode might not represent the* underlying *mode. Hence, the mode has to be estimated using the median.*

An important benefit of the truncated median filter, in addition to the removal of noise, is the implicit sharpening of edges [17]. As any neighbourhood approaches an edge, the distribution of that window becomes bimodal as in fig. 4.16. Replacing pixels close to edges by their modal values results in 'crisper' edges - a property which proves useful during subsequent edge analysis. Davies [17] also proved that there is negligible displacement of edges when the truncated median filter is applied. The same cannot be said for the orthodox median filter [24].

The truncated median filter was implemented using a 3x3 kernel. A single iteration was sufficient. The outermost 'noniles', containing a ninth of the distribution at each end, were considered as the extremes of the local distribution. This is because noniles were more stable than the minimum and maximum of the distribution. The sorting procedure of the median-finding loop was optimized for speed by applying

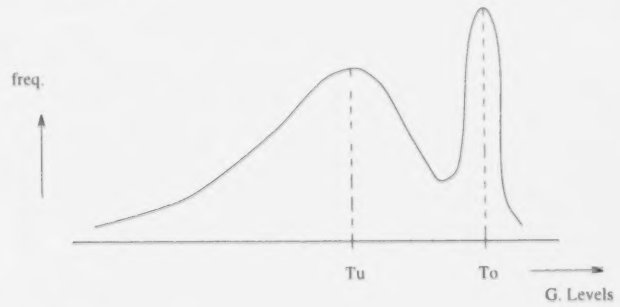

Figure 4.15: *Continuous analogue of fig. 4.14 where the mode of the distribution should be chosen as T_u (for underlying) rather than T_o (for orthodox) which has the highest frequency.*

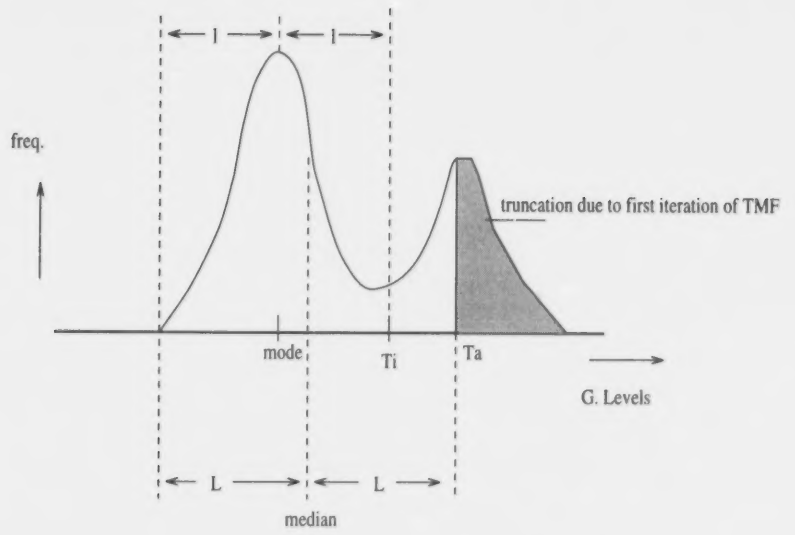

Figure 4.16: *Rationale for truncation. The obvious position for truncation is T_i. Since we use the median to guess at the position of the mode, T_a serves as a safe alternative to truncate this bimodal distribution. Iteration will draw T_a closer to T_i.*

the histogram sort. As pixel intensities must lie between 0 and 255, a histogram can be filled without having to scan the pixels for their boundary values (as in a general sorting routine). The algorithm then scans the histogram until the accumulated count of pixels reaches the medial value. For a 3x3 window, the median would be the intensity of the histogram bucket at the accumulated count of 5. The histogram sort has only some 256 operations for any size n. This proves to be more efficient than other general sorting operations [27].

Figures 4.17, 4.18 show an image (gamma corrected and resized) processed by the truncated median filter (TMF) just described. The reduction in spatial variations is evident from these images. To display the TMF's usefulness in removing edge noise zoomed portions of a low contrast, defocused image are observed. Figure 4.19 shows a zoomed image containing a triad-marker. Edges were determined without the application of the TMF (fig. 4.20). The edge image was noisy. The image in fig. 4.19 was then processed by the TMF. Fig. 4.21 is the filtered image. Fig. 4.22 shows the edges determined for the filtered image. The TMF image had a lower noise content than the unfiltered image in both gray levels and edges. Closer inspection showed

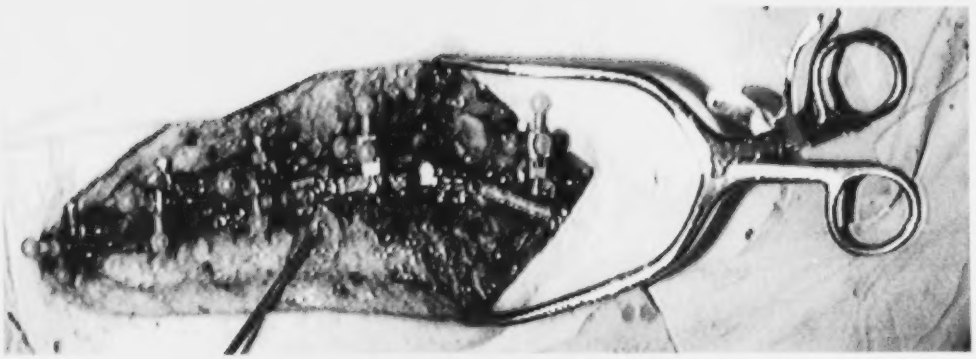

Figure 4.17: *A high contrast image before median filtering. Local entropy is high due to impulse noise and glare in the surgical FOV.*

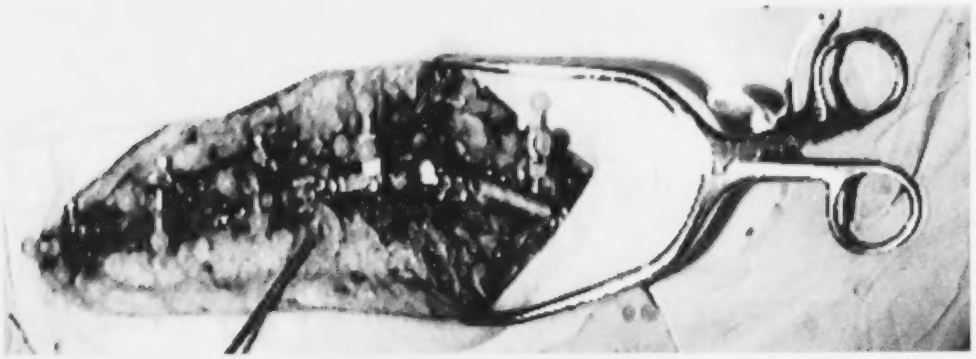

Figure 4.18: *The image (fig. 4.17) after truncated median filtering. Gray level variations are 'softer' with the removal of impulse noise. Edges are better defined and a consistent underlying gray level model emerges. TMF Parameters: 3x3 kernel, 1 iteration.*

Figure 4.19: *Zoomed portion of a low contrast image containing a triad. No truncated median filtering is applied here.*

that the filter also sharpened edges. Edges magnitudes were determined using a pair of $3x3$ Sobel masks [4].

4.2.4 Edge Detection

Edge detection is an essential part of most image processing applications. It has been applied to many problems such as image segmentation, feature location and threshold computation [28]. Edges provide the first level for abstraction where we can begin analyzing data based on intensity differentials and their directions. Data storage requirements and information processing are also substantially reduced in edge-space. There are two primary approaches to detecting edges. These are:

- Template Matching (TM)
- Differential Gradient (DG)

In either case, the aim is to find where the intensity gradient magnitude (g) is sufficiently large to be taken as a reliable indicator of the edge of an object. The two approaches differ mainly in how they estimate g locally; there are also differences in how they determine local edge orientations. Template matching applies a set of predefined masks, each with their own orientation, to a neighbourhood. The local edge gradient is chosen as the maximum of the applied masks. The orientation is assigned as the orientation of the chosen mask . Common examples of TM detectors are the Robinson masks and Canny edge detectors [28, 64]. The differential gradient approach relies on evaluating the edge response with respect to fixed cartesian directions. The magnitude and direction of the edge at a point is then evaluated vectorically (eq. 4.4, 4.5). If g_x and g_y are the edge responses in the cartesian directions, the local edge magnitude is calculated using the nonlinear transformation:

$$g = \sqrt{{g_x}^2 + {g_y}^2} \tag{4.4}$$

The edge direction, θ, is given by:

$$\theta = arctan(\frac{g_y}{g_x}) \tag{4.5}$$

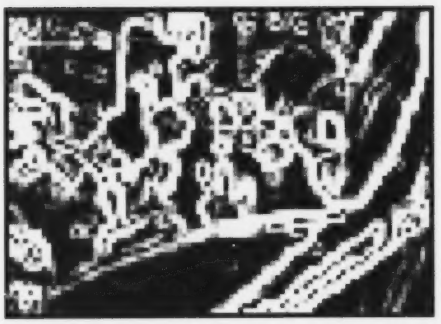

Figure 4.20: *The edge image of fig. 4.19. A 3x3 Sobel detector was used. The region of the image enclosed by primary edges is extremely noisy (compare fig. 4.22).*

Figure 4.21: *Fig. 4.19 processed by the 3x3 TMF.*

Figure 4.22: *The edge image of fig. 4.21. The TMF processed image is cleaner than its corresponding image (fig. 4.20, no TMF was applied).*

$$S_x = \begin{vmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{vmatrix} \qquad S_y = \begin{vmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{vmatrix}$$

Figure 4.23: *3x3 Sobel Edge Masks for the x, y directions*

It was found that correlation is more meaningful with edge images than with gray level images (refer pg. 55). Thus, a computationally efficient method to extract edges was required which would be part of the preprocessing stage of DETECT-POINT. The 3x3 Sobel detector (figure 4.23) is an ideal candidate for our application. As the Sobel detector is of the differential gradient type, there are only two masks involved and edge detection is speedy. Its directional accuracy has been quoted at one degree [18] and its performance as a general purpose edge detector outweighs that of other masks of the same class such as Roberts and Prewitt [53]. Template-based detectors produce greater accuracy but at a much higher computational cost. A typical operating room image transformed into the edge-space using the 3x3 Sobel masks is shown in figure 4.24.

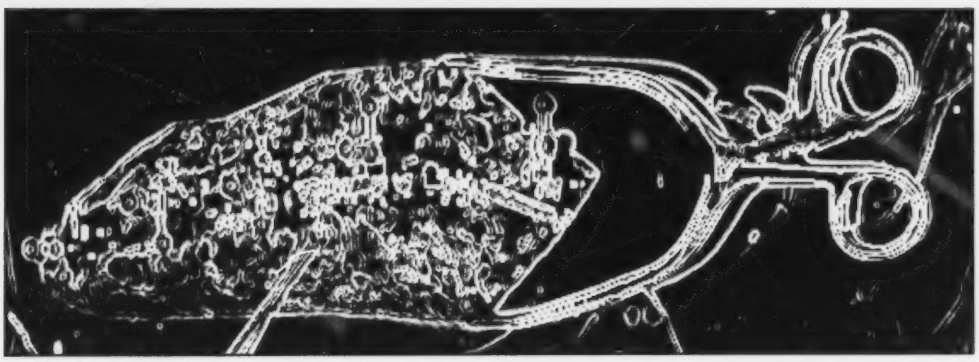

Figure 4.24: *OR image in edge-space (3x3 Sobel masks were used).*

4.3 Model-Based Recognition

Model-based recognition refers to a variety of algorithms that recognize the identity, position, and orientation of features in images using *a priori* knowledge about the feature's properties [13]. This approach was adopted in the form of template matching between statistically designed templates and real scenes.

The features to be identified in the operating room images are black, plastic, spherical structures which are part of markers mounted on vertebrae. Each marker contains three such spherical structures or 'ball-features' arranged in a known spatial configuration (fig. 4.1). Based on the analysis of operating room images (figs. 4.2, 4.3) with a number of feature location methods (pg. 30), it was decided that template matching would prove effective in locating these ball features. This is because:

- there is a small number of ball-features (between 6-21/image),
- all ball-features project as circular regions due to their spherical shape,
- the diameter of these circular regions is fixed at 10-11 pixels because of the range of markers from the cameras does not vary enough to affect feature size,
- each feature has a distinctive specular region due to light source between the cameras,
- there is significant contrast between most features and their background, and
- edge-space (fig. 4.24) images are more reliable than gray level space images because edges are more tolerant to photometric and perspective changes.

The template matching problem can be simply stated as: given an image, g(x,y) of size MxM, and given a template (model), t(x,y) of size NxN (where $N \ll M$ and the images are square for the simplicity of this definition), find the locations in g(x,y) whose neighbourhoods closely resemble t(x,y). Such a paradigm clearly involves three processes:

Similarity Measures Choosing or designing criteria to quantify similarity of a subimage with the template model

Template Design Design of templates for real images; organisation of these models for speed if there are multiple models

Feature Extraction Decision criteria for features based on results from the similarity criteria selected

The following sections discuss these processes in detail. Similarity measures for matching are considered first. This is followed by template design, and finally, decision criteria for feature selection.

4.3.1 Similarity Measures

By the above definition of the template matching problem, locations in the image, g(x,y) similar to the template t(x,y) must be found. This is visualized by taking the two-dimensional image t(x,y) and moving it over the image g(x,y) while computing the similarity with the image at each position. As the size, or domain, of t(x,y) is much smaller than that of g(x,y), at any position, it is only possible to evaluate the similarity of t(x,y) with a subimage g'(x,y), of the same size as the template, which lies directly below the template. This subimage, g'(x,y), is called the *shadow* of the template on the image. The similarity evaluation will be assigned to $g(x_0, y_0)$, the center of the shadow subimage. The image is searched by the template moving over the image in this way.

Similarity between the template and the shadow may be evaluated in many ways. This may be done by taking the sum of the differences between corresponding elements, as in distance measures, or the products between corresponding elements, as in correlation techniques. However dissimilar they may seem, these two methods are related. Consider the Euclidean distance, a common metric used to compare vectors and matrices. Using the Euclidean as our primary similarity metric between the image and a template, we have:

$$E(x,y) = \sqrt{\left(\sum_{i,j=-\frac{N-1}{2}}^{\frac{N-1}{2}} [g(x+i, y+j) - t(i,j)]^2 \right)} \tag{4.6}$$

or,

$$E_c(x,y) = \sqrt{\left(\sum_{i,j=}^{M-1}\sum_{0}^{} [g(i,j) - t(i-x, j-y)]^2\right)} \tag{4.7}$$

Note that eq. 4.6 is conventionally written in the form shown in eq. 4.7. In eq. 4.7, the indices i and j are bound by the limits of the image, g(x,y). The convention used in eq. 4.6 is followed in this work, where indices i and j are bound by the limits of the template, t(x,y) instead. It was found that correlation formulae were easier to assess when computed in the template's coordinate system, centered at $(i = j = 0)$, for any position of the template over the image. Mathematically, eq. 4.6 and eq. 4.7 are identical (proof in Appendix A).

The square of the measure in eq. 4.6 is given by

$$E^2(x,y) = \sum_i \sum_j \left[g^2(x+i, y+j) - 2g(x+i, y+j)t(i,j) + t^2(i,j)\right] \tag{4.8}$$

The popular cross-correlation function between two images is derived from the middle term of eq. 4.8 and is given explicitly in eq. 4.9. It is usually stated in this form although it is of limited use as a similarity measure as it is based on the assumption that the first and third terms of eq. 4.8 are constant. Those terms denote the energy of gray levels of the shadow and template subimages respectively. Of these, only the template energy is fixed, but the shadow's energy will vary from one position to another in any image.

$$C(x,y) = \sum_i \sum_j t(i,j)g(x+i, y+j) \tag{4.9}$$

To demonstrate the ineffectiveness of C(x,y), consider the calculation of C(x,y) in a part of an image with high energy, or high $g(x+i, y+j)$ values, the correlation value will be high irrespective of whether the template matches that area or not. To put eq. 4.9 in a more practical form, it is important to normalize it with the shadow's energy coefficient, $\hat{E}_{g'(x,y)} = \sum_i \sum_j g^2(x+i, y+j)$, where i,j are bound by the template size. This is done in eq. 4.10.

$$N(x,y) = \frac{\sum_i \sum_j t(i,j)g(x+i, y+j)}{\sqrt{\sum_i \sum_j g^2(x+i, y+j)}} \tag{4.10}$$

Besides being a good measure, the normalized cross-correlation index (eq. 4.10) can be modified to allow bounds on the value of C(x,y). By the Cauchy-Schwarz inequality [20],

$$\begin{aligned} N(x,y) &\leq \sqrt{\sum_i \sum_j [t^2(i,j)]^2} \\ N(x,y) &\geq -\sqrt{\sum_i \sum_j [t^2(i,j)]^2} \end{aligned} \tag{4.11}$$

Using eq. 4.11, the correlation coefficient can be modified to a very useful form.

$$\rho(x,y) = N(x,y) = \frac{\sum_i \sum_j t(i,j) g(x+i,y+j)}{\sqrt{\left(\sum_i \sum_j g^2(x+i,y+j)\right)\left(\sum_i \sum_j t^2(i,j)\right)}} \tag{4.12}$$

The correlation coefficient (ρ) now has the property of being bound by -1 and 1 (eq. 4.13). This property proves very useful when comparing correlation coefficients due to different templates when deciding the identity of the shadow.

$$-1.0 \leq \rho(x,y) \leq 1.0 \tag{4.13}$$

While the correlation coefficient of eq. 4.12 adjusts the underlying shadow for any scale changes it does not account for any differences in contrast between the shadow and template. This can be incorporated into eq. 4.12 by applying Moravec's zero-mean principle [47]. The zero mean principle states that in mask applications, best results are obtained when the mask and shadow have a mean value of zero. This can be applied to the above procedure by subtracting the mean of the neighbourhood from each pixel of the shadow before correlating with the template. The template may be adjusted for the mean only once. This leads to another form of the correlation coefficient given in eq. 4.14. Note that this form also conforms to the bounds specified in eq 4.13.

$$\rho(x,y) = \frac{\sum_{i,j=} \sum_{-\frac{N-1}{2}}^{\frac{N-1}{2}} \left(t(i,j) - \bar{t}\right)\left(g(x+i,y+j) - \bar{g}(x,y)\right)}{\sqrt{\left(\sum_{i,j=} \sum_{-\frac{N-1}{2}}^{\frac{N-1}{2}} \left(g(x+i,y+j) - \bar{g}(x,y)\right)^2\right)\left(\sum_{i,j=} \sum_{-\frac{N-1}{2}}^{\frac{N-1}{2}} \left(t(i,j) - \bar{t}\right)^2\right)}} \tag{4.14}$$

where,

$$\bar{t} = \frac{1}{N^2} \sum_{i,j=} \sum_{-\frac{N-1}{2}}^{\frac{N-1}{2}} t(i,j) \text{ , and}$$

$$\bar{g}(x,y) = \frac{1}{N^2} \sum_{i,j=} \sum_{-\frac{N-1}{2}}^{\frac{N-1}{2}} g(x+i, y+j) \tag{4.15}$$

As described herein, there are many forms of correlation, each with its own characteristics and computational requirements. As the different measures may be suited to different types of images, a quantitative assessment of the measures was required to decide on the optimal measure for OR images. It was felt that a study of the different forms of correlation was necessary. Previously, Boninsegna and Rossi [11] worked on the assessment of distance measures and Li and Madhavan [39] investigated measures for corner detection. Based on their professed functionality, five measures were chosen for testing on OR images. These were:

1. Euclidean Difference (ED) (eq. 4.6)
2. Orthodox Cross Correlation (OCC) (eq. 4.9)
3. Cross Correlation with Normalization Only (CCN) (eq. 4.10)
4. Cross Correlation with Means Adjustment Only (CCM) (eq. 4.16)

$$\rho(x,y) = \sum_{i,j=} \sum_{-\frac{N-1}{2}}^{\frac{N-1}{2}} (t(i,j) - \bar{t})\,(g(x+i, y+j) - \bar{g}(x,y)) \tag{4.16}$$

5. Cross Correlation with Normalization and Means Adjustment (RHO), or simply, the correlation coefficient (eq. 4.14)

An operating room image (fig. 4.25) and its corresponding edge magnitude image (fig. 4.32) were chosen as test images. The test templates (for gray levels and edges) were the averages of subimages at the 15 feature locations in another sample image (fig. 4.2). Basic statistics were gathered for all the above measures using both images (figures 4.31, 4.38). The response of the metrics was measured at manually specified feature locations of the test images and in a 50x50 subimage representative of the

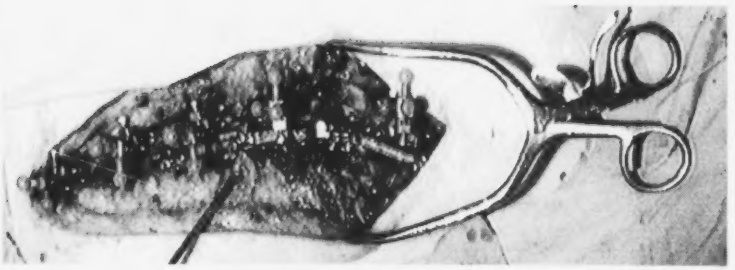

Figure 4.25: *Gray level test image.*

background in the decision spaces for each of the metrics. The decision space is the image with each pixel intensity replaced by the correlation coefficient, scaled to 0-255, at that point. Decision spaces for gray level space correlation are shown in figs. 4.26-4.30 and edge space correlation are shown in figs. 4.33- 4.37. A subimage of the surgical field-of-view was considered instead of the entire image because correlation statistics taken for the entire image are distorted by the response of the metric to the surgical drape which covers a large percentage of image area.

Based on figures 4.31 and 4.38, an index of performance called selectivity (ψ) was defined. Selectivity is the difference between the average response at feature locations (A_f) and background image patch(A_b) (eq. 4.17). As the decision spaces for the different measures were scaled to 0-255, the selectivities of different measures could be directly compared. Table 4.1 provides the results of our tests with a ranking of the measure-performances based on selectivity.

$$\psi = |A_f - A_b| \tag{4.17}$$

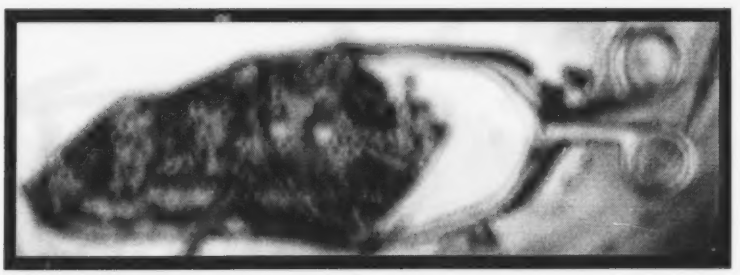

Figure 4.26: *Correlation of the gray level image (fig. 4.25) using the measure ED.*

Figure 4.27: *Correlation of the gray level image (fig. 4.25) using the measure OCC.*

Figure 4.28: *Correlation of the gray level image (fig. 4.25) using the measure NORM.*

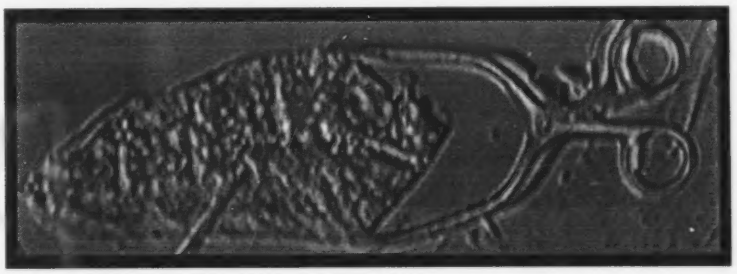

Figure 4.29: *Correlation of the gray level image (fig. 4.25) using the measure MEANS.*

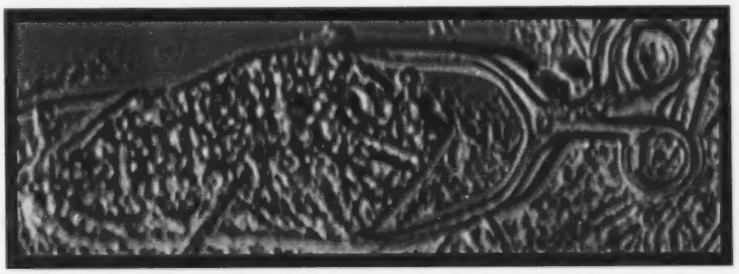

Figure 4.30: *Correlation of the gray level image (fig. 4.25) using the measure RHO.*

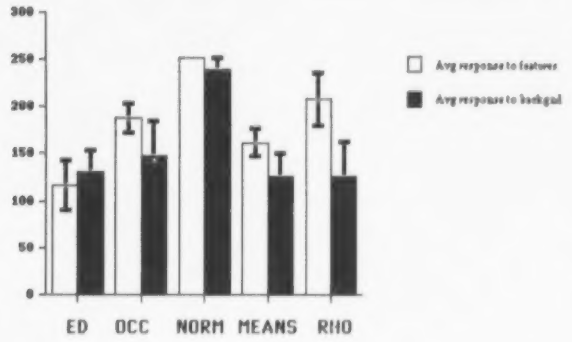

Figure 4.31: *Average Responses of different measures using gray level templates. The first bar denotes the average response to the 12 features in the image, the second bar denotes the average response to the 50 x50 subimage representative of the background.*

The performance of the measures differed with the property (gray level/edges) used for correlation. Using gray levels, the combined normalized cross-correlation measure (RHO) indicated the highest selectivity ($\psi = 81$). The other measures performed poorly in comparison (Table 4.1). Judging by selectivity, when edge images were used for correlation, there was significant improvement in the performance of CCM ($\psi = 44$ to $\psi = 60$) while RHO fell ($\psi = 81$ to $\psi = 72$). RHO, however, still performed better than CCM. The other measures performed poorly in comparison (Table 4.1). As expected, the overall performance of measures was better for correlation with edges when compared to gray levels.

These results confirm that RHO, the normalized cross-coefficient coefficient with means adjustment, is superior to other measures and independent of the property used for correlation. This is because it accounts for scale changes (using normalization), contrast differences (using means adjustment) and other perspective effects

	ED	OCC	CCN	CCM	RHO
Gray Levels	15	40	12	44	81
Edges	29	17	14	60	72

Table 4.1: *Selectivities of Measures. Selectivity = Avg. of Feature Responses - Avg. of response to Background*

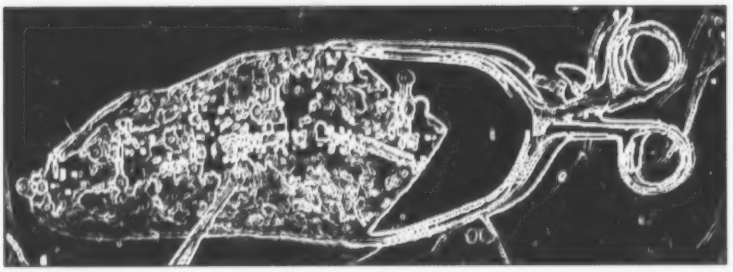

Figure 4.32: *The edge image corresponding to fig. 4.25.*

(like incompleteness, occlusion) and other photometric distortions in the image relative to the template. As RHO was the most robust operator with gray level images, it was selected for correlation with gray level images in DETECT-POINT.

Other than RHO, the only metric which performed reasonably was the cross-correlation measure with means adjustment only (CCM). As CCM involves less computation compared to RHO, but performs equally well with edge images (table 4.1), CCM was used for the correlation of edge images in DETECT-POINT.

CCN ranked lowest amongst the measures for both properties. In general, normalization without means adjustment (CCN) is of little value without the inclusion of means adjustment. Another notable point is that the OCC should not be overestimated based on its good performance ($\psi = 40$) with gray level images. This is because ball-features had a higher 'energy' level relative to their background and the OCC is somewhat only suited to such a situation. The OCC will always fail if the background image energy, $\hat{E} = \sum\sum g(x, y)$, is higher than that of the features themselves. It is less robust to distortions compared to other forms, like CCN, which is computationally close to the OCC.

4.3.2 Template Design and Organization

Template design is a challenging task when the model-based recognition process is to be applied to real images. The complexity of the problem is accentuated when we consider stereo images. The model or models developed must be robust enough to tolerate different perspective views, photometric distributions, occlusions and incom-

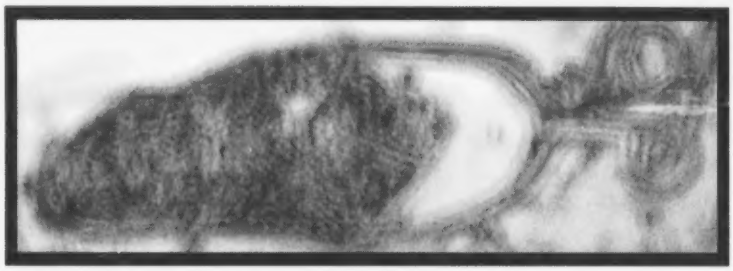

Figure 4.33: *Correlation of the edge image (fig. 4.32) using the measure ED.*

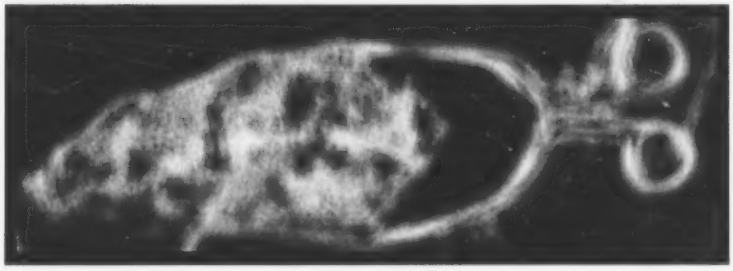

Figure 4.34: *Correlation of the edge image (fig. 4.32) using the measure OCC.*

Figure 4.35: *Correlation of the edge image (fig. 4.32) using the measure NORM.*

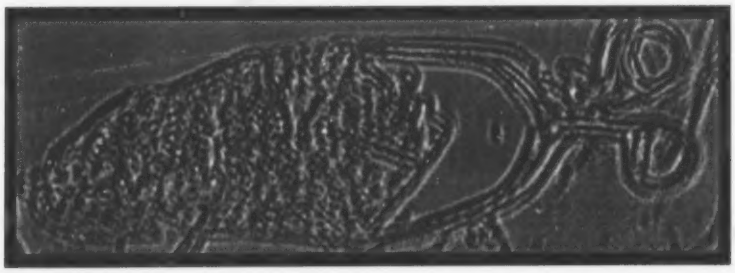

Figure 4.36: *Correlation of the edge image (fig. 4.32) using the measure MEANS.*

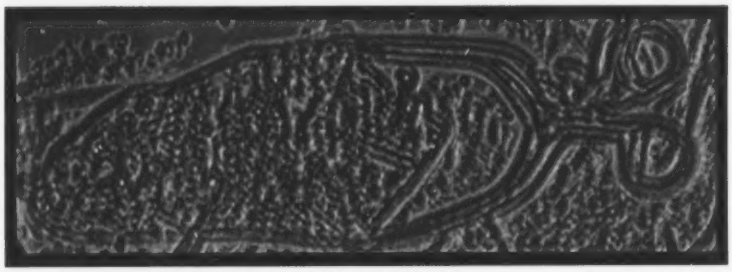

Figure 4.37: *Correlation of the edge image (fig. 4.32) using the measure RHO.*

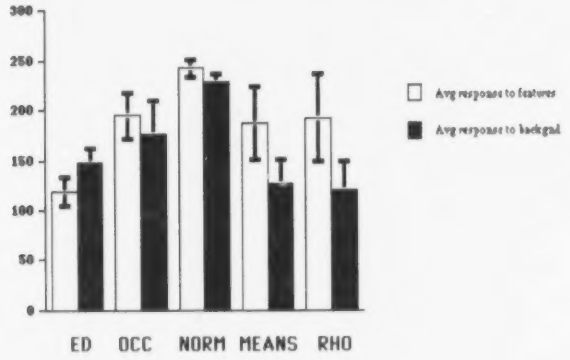

Figure 4.38: *Average Response of different measures using edge templates. Convention as in fig. 4.31.*

pleteness. Even though the correlation coefficient is robust, recognition will perform poorly if the models we develop are overly simplistic.

To deal with this problem, an early decision to use multiple models in different property spaces was made. Property spaces refer to representation of a scene on the basis of a particular property. For the operating room images, three property spaces showed promise:

Gray Level-Space where pixels represent the photometric distribution of the scene,

Edge-Space where pixels represent the edge magnitude as computed by the Sobel detector, and

Directivity-Space where pixels represent the edge direction as computed by the Sobel detector.

Only the first two were experimented with for this work. The third was left as an option for future work. The design and organisation of templates was done by taking a sample population of subimages, clustering the population into classes, designing representative templates for each class and organising these templates into an inverted tree-like structure. A template or subimage resides at each node of this tree (fig. 4.39, 4.42). A root template, designed using all the representative templates, occupies the root node of the tree while sample subimages, in their respective clusters, occupy the leaf nodes of the tree.

An approach similar to that attempted by Li *et. al.* [38] and Ramapriyan [54] was implemented. However, there are important differences. They organized their templates into a multilevel tree which was guided from root to leaf nodes by decision rules at each level of the tree. We have implemented a simpler tree structure with only two levels, the root template level and representative templates level. Hence, a single decision rule is required at the root template level: continue correlation (onto representative level) or reject tree. The pixel in the decision-space is assigned an optimal value based on correlation with representative templates. A multilevel tree would prove useful if all the ball-features were different; if they were spherical with different diameters, for instance. The experiments and results behind this work are now described.

Sample Population

A surgical case was photographed in three stages of correction called PRE, INTER and POST. From stage PRE, 27 subimages of size 19x19 were used to form the sample population of templates. There were 15 subimages from the left image (fig. 4.40) and 12 from the right image (fig. 4.41). Although there were 5 markers (15 ball-features) used during that surgery, 3 ball-features were occluded and completely invisible to the right camera. Hence, only 12 subimages were contributed by the right image. These subimages were taken from manually specified positions in the gray level-space. An equal number of subimages were obtained from the same positions in the edge-space. Ball-features occupied circular regions of approximately 10-11 pixels in the gray level images and 13-14 pixels in the edge magnitude images. The diameter is greater in the

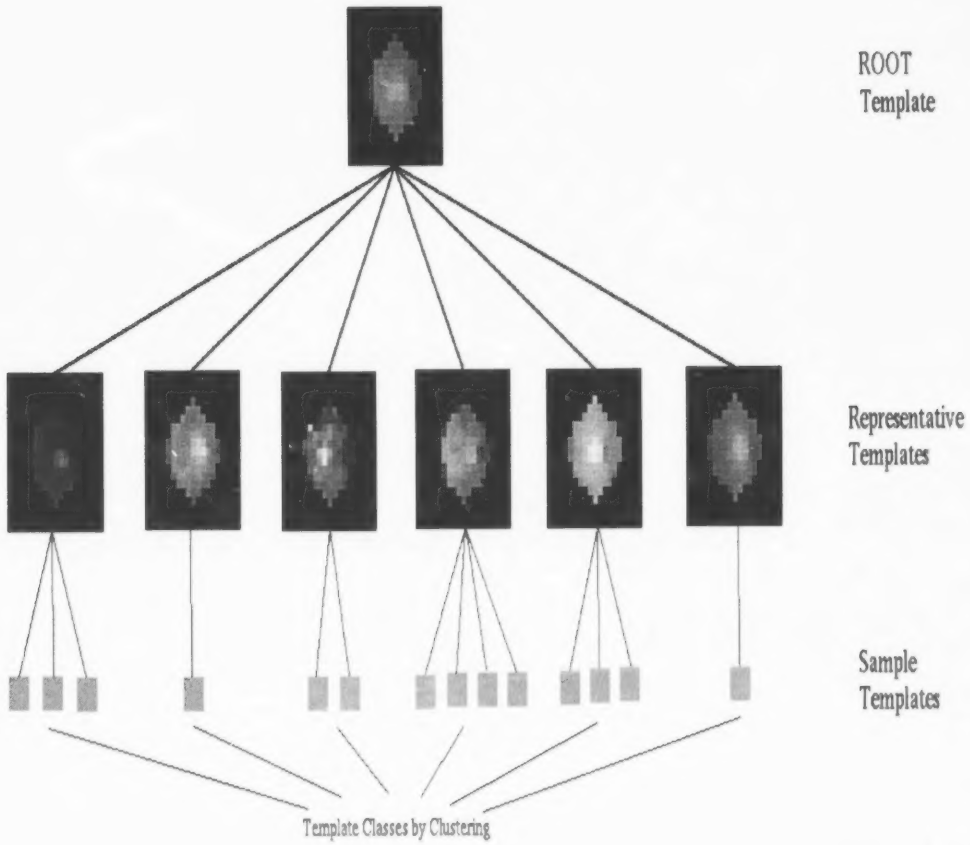

Figure 4.39: *Grey Level-Space Template Tree.*

edge-space because of smearing caused by edge detection. The smearing caused by directional masks is subject to thinning in many vision applications. As a smeared edge provides more information (area) for correlation than a thinned edge, thinning was not applied. The size of the templates was chosen as 19x19 to include background information. However, as the background was found to be highly inconsistent in both domains, only a thin annulus around the feature was considered for correlation in addition to the feature itself. Thus, correlation used circular templates rather than orthodox square/rectangular ones.

Clustering and Tree Generation

Clustering, in this context, refers to the arrangement of these subimages into classes where each class has some distinctive attributes. In pattern recognition, clustering is the technique used to define partitions or boundaries in feature space based on discriminant analysis [20] of a sample population of vectors. Hence, a nx1 vector population is treated in n dimensional space and the partitions are n-dimensional hyperplanes [20, 58].

Instead of orthodox clustering of 19x19 subimages, visual clustering was applied to the 27 subimages in each property space. In the gray level-space, the visual criteria included the position of the specular region within the ball's circular region, the absence of a specular region or a diffused specular region. In the edge level-space, the main criteria were the double rings, one for the ball-feature boundary and the other produced by the specular region. Classes were defined using the different positions of the smaller ring within the feature boundary. This procedure led to 6 classes of subimages in the gray level-space and 4 classes of subimages in the edge-space. Representative templates were generated for each class by averaging subimages in that class. Figures 4.39 and 4.42 show the representative templates for each of the classes in the two property spaces. The root template for each property space is the average of the representative templates of that space.

Clustering was quantitatively verified by constructing proximity matrices. These matrices compare the similarity between all pairs of representative templates to ensure that visual clustering has indeed resulted in dissimilar classes. The correlation

Figure 4.40: *Left Image: Stage PRE. 15 subimages of ball-features were obtained from this image.*

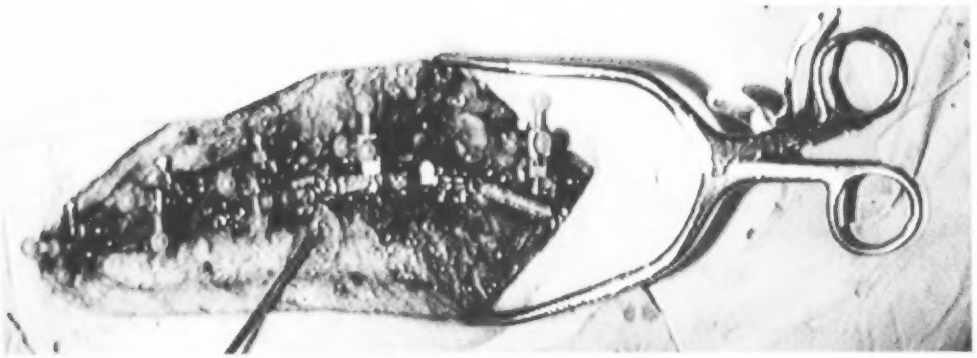

Figure 4.41: *Right Image: Stage PRE. 12 subimages of ball-features were obtained from this image.*

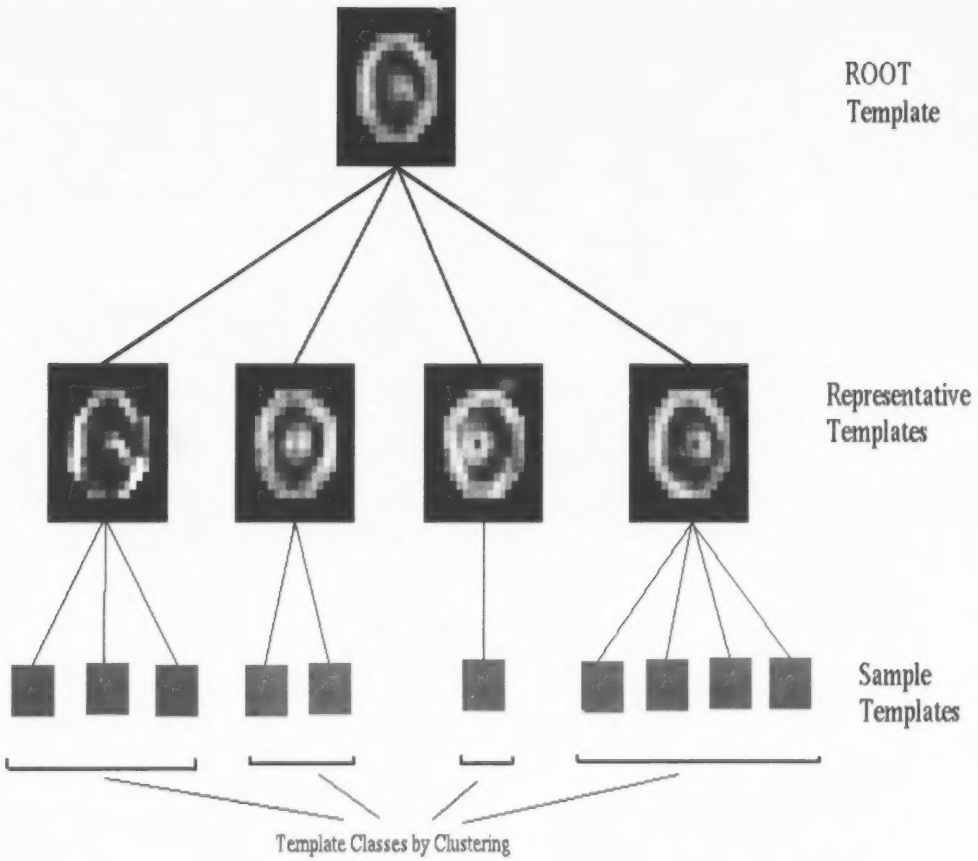

Figure 4.42: *Edge-Space Template Tree.*

	ROOT	RT#1	RT#2	RT#3	RT#4	RT#5	RT#6	Row Avg.
ROOT	1.000	0.975	0.995	0.981	0.985	0.995	0.995	0.9893
RT#1		1.000	0.961	0.957	0.961	0.959	0.958	0.9673
RT#2			1.000	0.966	0.977	0.991	0.997	0.9836
RT#3				1.000	0.956	0.973	0.964	0.9711
RT#4					1.000	0.970	0.974	0.9748
RT#5						1.000	0.994	0.9830
RT#6							1.000	0.9831

Table 4.2: *Proximity Matrix for the 6 Gray Level-Space Representative Templates.*

	ROOT	RT#1	RT#2	RT#3	RT#4	Row Avg
ROOT	1.000	0.951	0.975	0.963	0.974	0.9727
RT#1		1.000	0.897	0.870	0.914	0.9261
RT#2			1.000	0.933	0.938	0.9486
RT#3				1.000	0.912	0.9356
RT#4					1.000	0.9475

Table 4.3: *Proximity Matrix for the 4 Edge-Space Representative Templates.*

coefficient was used as the proximity criterion for the construction of these matrices. The 6 classes in the gray level-space led to the generation of a 6x6 proximity matrix (table 4.2) between the representative templates of the classes. Similarly, the edge-space had a 4x4 proximity matrix for its 4 classes (table 4.3). Li *et. al.* [38] used such matrices with different proximity criteria to generate multilevel trees as part of the complete link clustering algorithm. We used the same methods for verification of visual clustering. The proximity matrices have elements with large values corresponding to the similarity between different representative templates because the templates represent different imaged appearances of the same object. Proximity matrices of subimages within any of the classes produce elements which are all approximately 1 (correlation coefficient, $\rho \approx 1.0$). The proximity between subimages in the classes is thus greater than that between representative templates of the different classes. This is an indication that visual clustering was successful.

4.3.3 Feature Extraction

This section deals with the use of previously designed template trees for template matching on operating room images. The procedures used to handle multiple decision spaces to extract sub-pixel resolution features are described. Unlike a property space (PS) which corresponds to a physical property like gray level or edge magnitude of a scene, a decision space (DS) corresponding to a PS represents the result of correlation in that PS. Once gray level and edge images have been processed, decisions need to be made regarding the positions of features; hence the name decision space.

Decision Spaces

The value of a pixel in the decision space (DS) depends on the neighbourhood of that pixel in property space (PS) and the template tree for that PS. If the neighbourhood of a point, $P_{DS}(x,y)$, in a DS is close to the root template, it is assigned the maximum of the correlation coefficients on correlation with the representative templates of the corresponding PS (eq. 4.18) . There is only one threshold required at the root level. This threshold can be set conservatively without affecting the accuracy of feature detection. This is because it only saves the unnecessary effort of correlation of each point's neighbourhood with all representative templates.

For a property space, the corresponding decision space, $P_{DS}(x,y)$, is given as:

$$\begin{aligned} P_{DS}(x,y) &= 0.0 && \text{if } \rho^R(x,y) < T \\ &= \max_i[\rho^i(x,y)] && \text{otherwise} \end{aligned} \tag{4.18}$$

where $P_{DS}(x,y)$ is the DS value at location (x,y), T is the threshold which allows/disallows correlation with representative templates, $\rho^i(x,y)$ is the correlation coefficient of the PS shadow image at (x,y) with the i*th* representative template , and $\rho^R(x,y)$ is the correlation coefficient with the root template.

The gray level PS in figure 4.43 was processed with the gray level template tree (fig. 4.39). The corresponding DS, scaled to a [0-255] range, is shown in figure 4.44. For the edge-space image (fig. 4.45), the corresponding scaled DS is shown in figure 4.46.

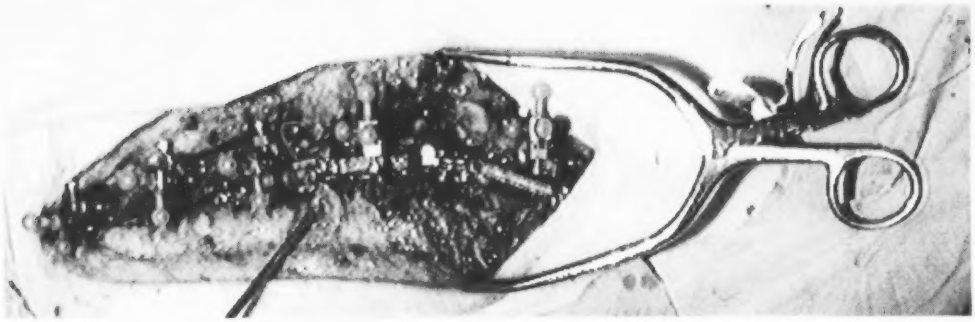

Figure 4.43: *The Gray level Property Space.*

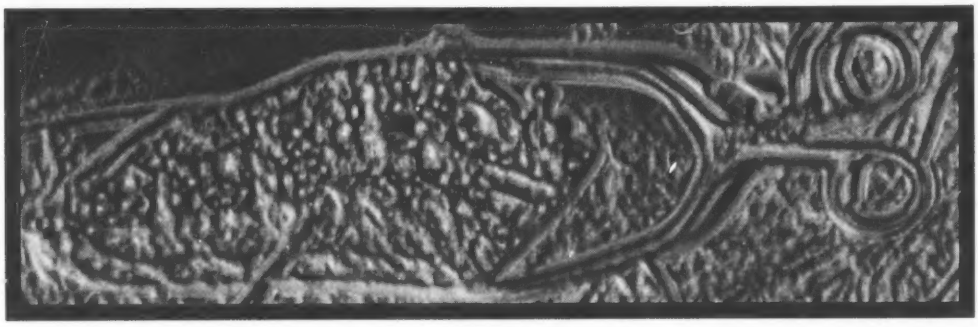

Figure 4.44: *Decision Space (scaled to 0-255) corresponding correlation of fig. 4.43.*

Feature Space

As correlation results in each PS are represented in a DS, the number of decision spaces is equal to the number of property spaces selected. The software developed for processing operating room images, DETECT-POINT, uses two such properties, gray level and edge magnitudes. The results in these DS (figs. 4.44, 4.46) cannot be used directly to extract features. Information from multiple DSs must be appropriately interpreted and represented in a single place. This is done by constructing the feature space (FS). Feature locations are derived directly by manipulating the FS only.

The edge DS (fig. 4.46) contains features which can be distinguished by the *non-maximal suppression* [32] in their neighbourhoods. This property is not exhibited in the gray level DS (fig. 4.44) where feature locations are surrounded by a 'bulbous' neighbourhood where it is difficult to localize the feature. It was found that the edge DS conveys more information than the gray level DS. This could be quantitatively proven using point spread functions for feature locations in both decision spaces.

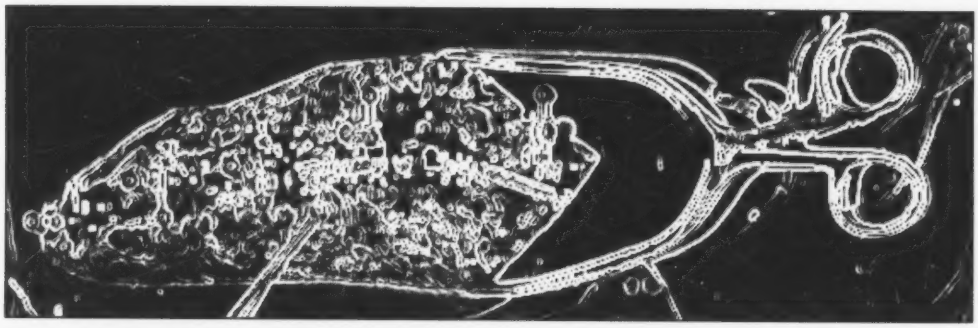

Figure 4.45: *The Edge Magnitude Property Space.*

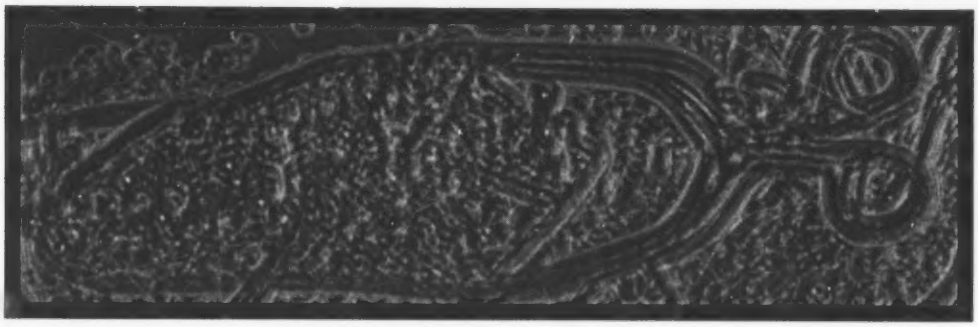

Figure 4.46: *Decision Space (scaled to 0-255) corresponding to correlation of fig. 4.45.*

This is to be expected as edges are reliable while gray levels differ significantly due to differences in camera perspectives to a scene. It can be argued that since the edge image is derived from the gray level image, the edge image will never be isolated from photometric distortions. However, these distortions will be less apparent in the edge PS as compared to the gray level PS. While the edge PS seems to be superior in this application, another application might have property spaces where each PS has its pros and cons. Hence, each application will differ in the rules for the construction of the feature space from the different decision spaces. In this application, the feature space is constructed (eq. 4.19) by weighting the edge DS twice as much as the gray level DS. The factor of 2 was chosen *ad hoc* but serves the purpose of emphasizing edge information more than gray level information.

$$P_F(x,y) = \frac{P_G(x,y) + K * P_E(x,y)}{K+1} \tag{4.19}$$

where K is the weighting factor (K=2 was used) and P_F, P_G, P_E denote the value of a pixel position (x,y), in feature space, gray level DS and edge DS respectively.

Segmentation

Segmentation of the feature space (FS) is the process of distinguishing feature pixels (signal) from noise. A histogram-based, percentile-like method [57] was used for dynamic thresholding. Based on the number of features expected, a maximum number of pixels admissible as 'feature pixels' is calculated. A histogram of the FS is constructed. The intensity buckets of the histogram are accumulated starting at 255. The threshold is selected as the intensity at which the accumulated count exceeds the maximum number of admissible pixels (eq. 4.20). Thresholding is done so that pixels passing the threshold maintain their FS values.

$$\sum_{i=255}^{j} < MAX \tag{4.20}$$

where, j is the threshold and MAX is the maximum number of admissible pixels,

On thresholding, the FS corresponding to figures 4.43 and 4.45 is shown in figure 4.47. It was known *a priori* that there are 12 ball-features in the scene. Allowing 6 pixels per feature, the upper 72 pixels from the histogram of the FS survive thresholding. After thresholding (figure 4.47), the FS image is subjected to a two-pass algorithm to search for and label all 8-connected components [28]. Each connected component is assumed to represent an extracted feature; a possible ball-feature location. An intensity-weighted centroid calculation for each connected component yields the subpixel location for the feature (eq. 4.21, 4.22).

Within a connected component (comp.), the feature location (F_x, F_y) is calculated as:

$$F_x = \frac{\sum\sum_i^{over\ comp.} G_i x_i}{\sum\sum_i G_i} \tag{4.21}$$

$$F_y = \frac{\sum\sum_i^{over\ comp.} G_i y_i}{\sum\sum_i G_i} \tag{4.22}$$

where (x_i, y_i) is a member of the connected component and has intensity G_i.

Figure 4.47: *Feature Space after dynamic thresholding (K=2, 6 pixels/component).*

4.3.4 Results

As mentioned in an earlier section (pg. 59), templates were designed using stage PRE of a selected surgery. After clustering, there were 6 representative templates in gray level-space and 4 representative templates in edge-space. The software developed for the task of ball-feature detection was called DETECT-POINT. The accuracy of DETECT-POINT was tested against manually digitized ball-features. As manual digitization itself is only accurate to 0.5 pixels (considering integral pixel values only), the sub-pixel accuracy of DETECT-POINT is difficult to assess in absolute terms. Instead, the completeness and reliability of the software were tested in all three stages (PRE, INTER, POST) of the selected surgery. DETECT-POINT returned *all* manually detected ball-feature locations to within 1 pixel of their manually digitized locations. Considering the inaccuracy inherent in manual digitization, the worst case accuracy of DETECT-POINT is 1.5 pixels, 0.5 pixels more than the difference between manually and automatically detected features (table 4.4). Table 4.4 compares subpixel locations of features returned by DETECT-POINT against the corresponding manually detected positions. An average of 30-35 features were returned for every OR image. As there were 15 features, there is approximately one false positive located for every feature in the scene.

An assessment of template matching would be unfair without addressing its computational and memory requirements. Even among low level image processing methods, correlation is one of the most expensive in computational terms. More so in this

application, where multiple templates are applied in multiple domains (gray levels, edges). It is difficult to apply such a method to a true real-time application without resorting to specialized or parallel architectures. DETECT-POINT took an average of 240 seconds to process a stereo-image on an IBM RS/6000. This was, however, a spatial domain implementation of correlation. It was estimated that an FFT-based implementation for correlation in the spatial frequency domain will lead to processing times in the range of 10 seconds (for large images and templates), suitable for use in a pseudo-real time environment like the OR. However, only the OCC form of cross-correlation, which was proved to be of little use, is amenable to a direct application of the FFT. A formulation is presented in Appendix A for the conversion of the most involved form (RHO) into subparts where the FFT can be applied. The proofs presented in the appendix can be applied to make any of the other forms (CCM, CCN, ESM) amenable to FFT implementation.

In summary, a feature detection algorithm has been developed for the automatic recognition of the ball-components of triad markers in OR images. Preprocessing routines were developed to standardize raw images. They consisted of gamma correction, truncated median filtering, edge detection, and a novel size reduction algorithm. Model-based recognition was implemented as a template matching problem in multiple property spaces with statistically designed templates. Five similarity measures, different forms of cross-correlation, were investigated. The measure RHO proved superior to the others when correlation used gray levels and edges. A formulation of RHO suitable for FFT implementation was also done (Appendix A). Templates for each property space were organized in a tree. Feature extraction resulted in the detection of *all* visible features in the 6 images from a surgery. The average error in the location of these features was one pixel from their manually detected locations. There was an average of one false positive for every feature in the image. The feature sets obtained by the detection process are used as inputs to higher level correspondence processes along with calibration parameters. These are responsible for the elimination of false positives based on other constraints. The performance of this approach to feature detection is comparable to others [47] which require detection for the purpose

Manual Detection		Automatic Detection		Error		Euclidean Error
X_m	Y_m	X_a	Y_a	$\Delta X = X_m - X_a$	$\Delta Y = Y_m - Y_a$	$\Delta = \sqrt{(\Delta X)^2 + (\Delta Y)^2}$
15	121	15.00	121.00	0.00	0.00	0.00
34	135	33.00	135.48	1.00	-0.48	1.11
71	97	70.83	96.83	0.17	0.17	0.24
91	118	90.69	118.34	0.31	-0.34	0.46
130	90	129.81	88.60	0.19	1.40	1.41
149	109	148.99	109.25	0.01	-0.25	0.25
213	48	212.00	49.01	1.00	-1.01	1.42
215	73	213.00	72.53	2.00	0.47	2.05
199	71	198.66	70.33	0.34	0.64	0.72
317	48	315.00	48.00	2.00	0.00	2.00
317	69	317.00	68.52	0.00	0.48	0.48
300	70	299.00	69.00	1.00	1.00	1.41
Average Error (pix.)				0.67	0.17	0.96
Standard Deviation of Error (pix.)				0.70	0.64	0.67
Number of False Positives					17	

Table 4.4: *Manual versus Automatic Detection (stage POST, all values in pixels).*

of reconstructing in 3D rather than identifying objects only (eg. mammogram analysis [35]). The next chapter deals with the algorithms developed for the registration of feature sets returned by automatic detection on stereo-images. Correct registration enables the system to reconstruct points in 3D and identify triads corresponding to the ones used during surgery.

5 Hierarchical Feature Correspondence

Hierarchical feature correspondence in the OR stereo-vision system refers to the registration of features in layered stages where each stage uses the results of the previous one. While the lower stages operate on 2D image features, finding candidates for correspondence in the other image of the stereo-pair, higher stages function in 3D, configuring sets of 3D points to *a priori* models (fig. 5.1).

This chapter deals with issues related to correspondence between features located in the left and right images. Reliable correspondence between stereo-views, an integral component of the stereo paradigm [5], is a challenging and active field of research today. This is because while the other components of the stereo paradigm like 3D reconstruction are quantitatively well defined [59], correspondence of pixels or features between perspectively different views is not. There is a one-to-many relationship between each image pixel and the line in object-space it corresponds to. Different perspective views of three dimensional scenes present different issues. The field of view (FOV) of the stereo-images will differ so that some points in one view will be 'outside' the other. Other problems include the projection of object-space points such that both views have neighbourhoods with different gray level or structural properties (eg. edges), the basis of image-related inference. Occlusion, where a point in one scene is 'hidden' in the other, is another issue which must be addressed by correspondence algorithms or minimized by system design.

The organization of this chapter is as follows. First, a literature review of correspondence as a problem in stereo-vision is presented. It includes an outline of the hierarchical strategy developed for correspondence between features in OR stereo-images. Subsequent sections deal with the three main stages of this strategy. Finally, a section on simulations and results is provided. Simulations are included to point out the limitations of the camera model and to justify the use of constraints and heuristics used to solve the correspondence problem with real data.

5.1 Literature Review

The correspondence problem in a feature-based stereo approach [30] uses scene and camera-geometry specific information with a number of constraints for the identification of corresponding or matching features in stereo-images. The purpose of feature or pixel correspondence allows calibration and/or reconstruction in three dimensional object-space [22]. For a generalized scene, where little information is available about the scene and cameras or where incomplete information is used to develop models of the immediate environment, there are no robust solutions [60]. Robotic systems tend to use generalized *interest operators* to detect *interesting* image points to construct a model of the 3D world surrounding the robot. The interest operator by Moravec [43, 47] was based on directional variance. The assumption underlying these approaches is that a set of 3D points will project as interesting points in both views and can be reconstructed to yield a sparse estimate (model) of the robot's environment. Reconstructing a model of the spine from stereo-images of the surgical field of view only falls into an even more complex class of problems, as much of the object of interest, the spine, is not even visible. The complexity of the OR scenes was reduced by rigidly fixing locatable markers onto selected vertebra. These markers were located using correlation-based methods and statistically designed template models. The process of correspondence for this application was thus simplified to matching detected ball-components and identifying the triads they belong to.

Constellation matching (CM) [8] was investigated thoroughly as a possible solution to matching during a pilot study. Three promising methods were found. These used angular world view by Murtagh [49], relaxation by Ranade and Rozenfeld [55] and tree search by Wong [72]. For the pilot study , Wong's method using a tree search with branch and bound heuristics was successfully implemented for spine-model images. However, with OR images, this approach proved inadequate. This was because CM is directed towards matching 2D point configurations where the stereo-images can be considered to be translated and rotated versions of the same 2D configuration. CM proved successful in the pilot study with a 3D point configuration because of

the smooth variation of depth for neighbouring point features and the near-parallel (small baseline) stereo-geometry used.

Other generalized, more flexible algorithms were studied. The cooperative stereo paradigm by Marr and Poggio [41] introduced the concept of disparity, or the parallax/shift between features in stereo-images. Recent approaches [6] have used general assumptions like smoothness of the imaged surface, rigidity of sub-feature components of features, and uniqueness of feature matches to model the behavior of disparity and design constraints for feature correspondence. The disparity gradient limit of the Pollard, Mathew, and Frisby algorithm, better known as PMF [52], used cyclopean constraints based on biological knowledge of human stereoscopic vision. This approach is especially useful when camera geometry is undetermined. Most researchers integrate these approaches and adapt them to specific stereo problems. The formulation of feature detection and matching as a special case of signal detection by Wu and Ahuja [32] is one example of the integration of different constraints to solve for correspondence.

5.2 Correspondence Strategy

The OR problem, as a whole, is not as ill-conditioned as many of the situations addressed by the above referenced research. In this case, specific triad-markers are planted for easy detection, a well defined model existed for the configuration of the ball-feature components of triad-markers, and calibrated cameras were used for accurate reconstructions. Also, system-related information about the OR layout is available. For instance, the approximate range and the number of triad-markers used during each surgery is known. While the OR problem is more controlled than the generalized correspondence problems mentioned earlier, many of the assumptions made for the solutions to the generalized problems cannot be applied to the OR case. Constellation matching, which proved successful for feature correspondence during a

pilot study, is inappropriate because of the significant vergence[1] and large baseline[2] of the OR camera system. Disparity constraints based on smoothness cannot be applied as neighbouring features (ball-components) cannot be considered as belonging to any particular smooth surface. Because of the lack of such constraints, the uniqueness constraint was not applied on feature pixels until the final stage of identifying triad-markers.

The strategy developed herein for the registration of detected ball-features between stereo-images was aimed at using scene and system-specific information before resorting to more generalized constraints. This led to the hierarchical structure of this approach (fig. 5.1). The algorithm processed feature sets for the stereo-pair provided by the detection routines described earlier. At the lowest level, each feature was considered with all the features in the other image of the stereo-pair. Constraints based on epipolar geometry, system range, and disparity were used to construct initial sets of candidates. The next higher level attempts to refine these candidate sets using structural similarity in the scene. All candidates in the 'refined' candidate sets are then reconstructed. This results in reconstruction of multiple object-space points (3D features) for image (2D) features whose refined candidate sets contain multiple members. At the highest level of the correspondence hierarchy, features in 3D are tested for configurations matching the 3D triad-marker models. Heuristics were used to discard false triads and reconstructed points so that a unique number of triads, corresponding to the number of triads used in that experiment or surgery, survive. The subsequent sections describe the different stages of this hierarchical method in detail. Results are shown at the end.

5.3 Candidate Sets

Given a feature set for the left stereo-image (FS_L), the first task of correspondence was to select, for every feature in this set, a set of candidate points from the right stereo-

[1]Vergence refers to the angular convergence between stereo cameras. Parallel geometries, for example, have no vergence associated with them.

[2]Baseline refers to the distance between stereo-cameras.

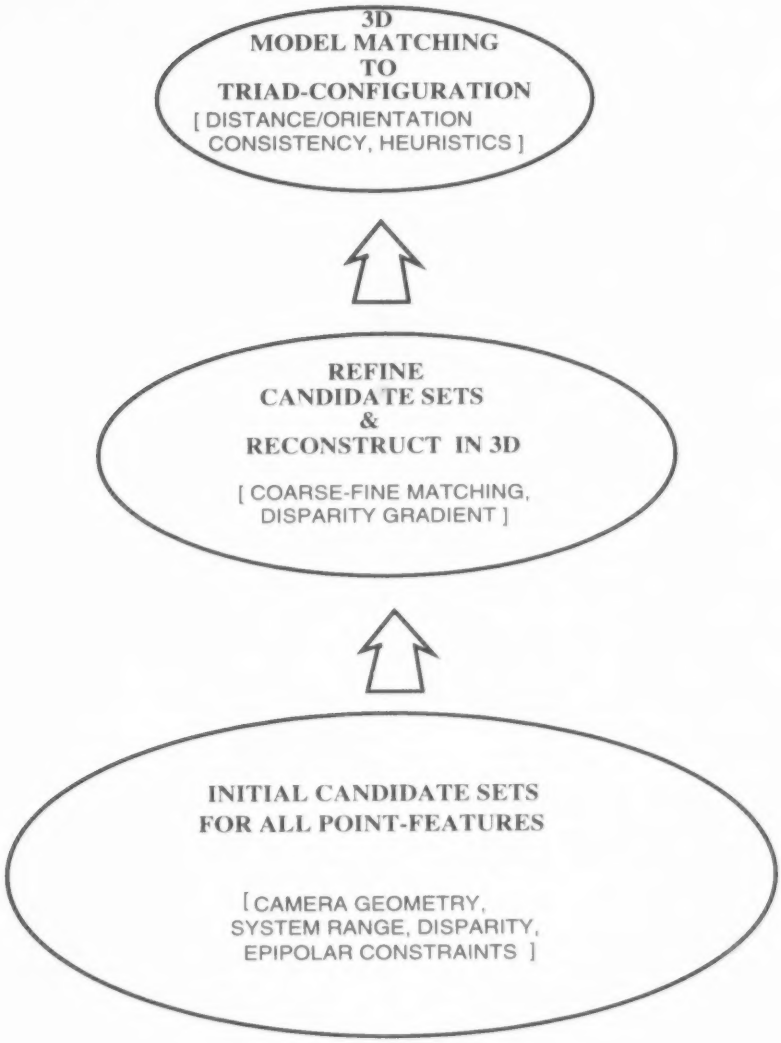

Figure 5.1: *Hierarchical Correspondence Paradigm for the OR system.*

image feature set (FS_R) which definitely included the left-image feature's 'match' in the right-image. This was called left to right registration. Similarly, right to left registration involved the assignment of subsets of the left-image feature set as candidate sets to features in the right image.

A description of the left to right registration process for a point-feature in the left image (P_L) will serve to describe the algorithm used for constructing candidate sets. The right to left registration process occurs in an identical fashion. There were four conditions used for constructing the candidate set (CS_{P_L}) for P_L from the feature set for the right image (FS_R). These were (also refer to figs. 5.2, 5.3):

Range Constraint Determine the 3D close (P_c^L) and far (P_f^L) points in the left camera coordinate system for image point P_L corresponding to system range boundaries, [R_{min}, Rmax],

Camera Geometry Constraint Represent the close and far points (P_c^L, P_f^L)in the right camera coordinate system (P_c^R, P_f^R),

Collinearity Condition Perspectively project P_c^R and P_f^R onto points c and f respectively on the right image plane to obtain the epipolar line segment, c-f, corresponding to P_L, and

Epipolar Constraint Points from the right feature set, FS_R, which are perpendicularly close to the epipolar segment c-f are accepted into the candidate set, CS_{P_L}, of P_L.

The process of constructing candidate sets is now described using the constraints mentioned above. The use of left to right registration for feature P_L is continued for the rest of this section.

Range Constraint

In the left camera coordinate system, the range at which markers were located was specified by the limits $Z \in [Z_c, Z_f]$, where Z_c and Z_f are positive. Z_c and Z_f were trignometrically derived as functions of the vergence, baseline and range boundaries of the stereo-cameras (fig. 5.3).

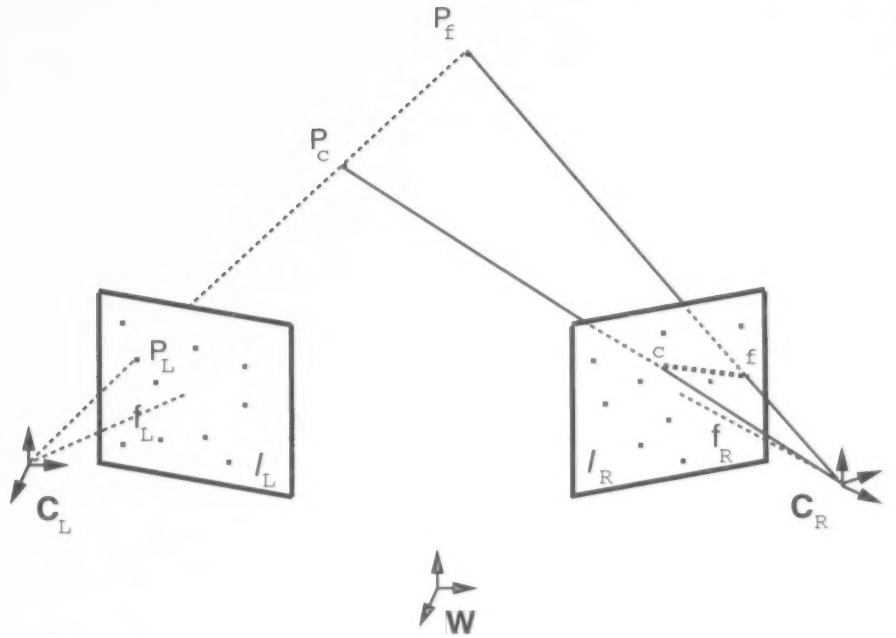

Figure 5.2: *Determining the epipolar segment (c-f) using camera geometry and range.*

Given the left image feature, $P_L(x_l, y_l)$, collinearity was used to determine its 3D close point, $P_c^L(X_L,Y_L,Z_L)$ (fig 5.2), where $Z_L = -Z_c$, negative of the close point range (all object-space points have negative z-coordinates in the camera coordinate system).

$$
\begin{aligned}
\overline{C_L P_L} &= k\overline{C_L P_c}, \\
\begin{bmatrix} x_l \\ y_l \\ -f_L \end{bmatrix} &= k \begin{bmatrix} X_L \\ Y_L \\ -Z_c \end{bmatrix}, \\
\frac{x_l}{-f_L} = \frac{X_L}{-Z_c} \quad &\text{and} \quad \frac{y_l}{-f_L} = \frac{Y_L}{-Z_c}, \\
P_c^L(X_L, Y_L, Z_L) &= \left(-\frac{x_l}{f_L} Z_c, -\frac{y_l}{f_L} Z_c, -Z_c \right)
\end{aligned}
\tag{5.1}
$$

The coordinates of the far point of P_L in the left camera system (P_f^L) are also determined in this manner.

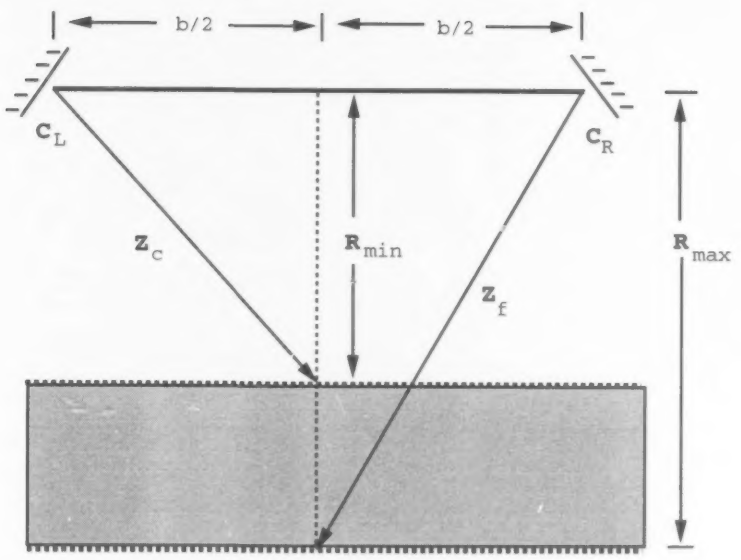

Figure 5.3: *Calculation of 3D close and far points using baseline and range boundaries.*

Camera Geometry Constraint

As the location and attitude of the cameras are known from calibration, any point in the left camera's coordinate system can be transformed to the right camera's coordinate system using eq. 5.2.

$$\begin{bmatrix} X_R \\ Y_R \\ Z_R \end{bmatrix} = \bar{\mathbf{R}}_{LR} \cdot \begin{bmatrix} X_L \\ Y_L \\ Z_L \end{bmatrix} + \bar{\mathbf{T}}_{LR} \tag{5.2}$$

where, (X_L, Y_L, Z_L) and (X_R, Y_R, Z_R) are coordinates of the same point in the left and right stereo-camera coordinate systems respectively; $\bar{\mathbf{R}}_{LR}$ and $\bar{\mathbf{T}}_{LR}$ are the rotational and translational matrices relating the left camera system to the right camera system. It is now shown how these matrices can be determined using the extrinsic parameters of the stereo-cameras.

From collinearity (eq. 3.3), the object-space coordinates (X_W, Y_W, Z_W) of any point can be transformed into 3D coordinates in a camera system (X_C, Y_C, Z_C) using

eq. 5.3.

$$\begin{bmatrix} X_C \\ Y_C \\ Z_C \end{bmatrix} = \bar{\mathbf{M}}. \begin{bmatrix} X_W - X_0 \\ Y_W - Y_0 \\ Z_W - Z_0 \end{bmatrix}$$

or,

$$\bar{\mathbf{X}}_C = \bar{\mathbf{R}}_C \left(\bar{\mathbf{X}}_W - \bar{\mathbf{T}}_C \right) \tag{5.3}$$

where, the camera's rotational matrix, $\bar{\mathbf{R}}_C = \bar{\mathbf{M}}$, and its translational vector, $\bar{\mathbf{T}}_C = [X_0\ Y_0\ Z_0]^T$; $[X_0\ Y_0\ Z_0]^T$ are the object-space coordinates of the camera (perspective center), and $\bar{\mathbf{M}}$ is a function of the rotational angles (ω, ϕ, κ) (eq. 5.4) of the camera considered in that order [46, Appendix A].

$$\bar{\mathbf{M}} = \bar{\mathbf{M}}_\kappa \bar{\mathbf{M}}_\phi \bar{\mathbf{M}}_\omega = \begin{bmatrix} \cos\phi\cos\kappa & \cos\omega\sin\kappa + \sin\omega\sin\phi\cos\kappa & \sin\omega\sin\kappa - \cos\omega\sin\phi\cos\kappa \\ -\cos\phi\sin\kappa & \cos\omega\cos\kappa - \sin\omega\sin\phi\sin\kappa & \sin\omega\cos\kappa + \cos\omega\sin\ phi\sin\kappa \\ \sin\phi & -\sin\omega\cos\phi & \cos\omega\cos\phi \end{bmatrix} \tag{5.4}$$

For the left and right cameras, writing eq. 5.3 in matrix form, we get

$$\begin{aligned} \bar{\mathbf{X}}_L &= \bar{\mathbf{R}}_L \left(\bar{\mathbf{X}}_W - \bar{\mathbf{T}}_L \right) \\ \bar{\mathbf{X}}_R &= \bar{\mathbf{R}}_R \left(\bar{\mathbf{X}}_W - \bar{\mathbf{T}}_R \right) \end{aligned} \tag{5.5}$$

Eliminating $\bar{\mathbf{X}}_W$ from (5.5), we can relate the coordinates of $\bar{X}_W$ in both camera coordinate systems,

$$\bar{\mathbf{X}}_R = \left(\bar{\mathbf{R}}_R \bar{\mathbf{R}}_L^{-1} \right) \bar{\mathbf{X}}_L + \bar{\mathbf{R}}_R \left(\bar{\mathbf{T}}_L - \bar{\mathbf{T}}_R \right) \tag{5.6}$$

where object-space point $(\bar{\mathbf{X}}_W)$ is given by $\bar{\mathbf{X}}_L$ and $\bar{\mathbf{X}}_R$ in the left $(\bar{\mathbf{R}}_L, \bar{\mathbf{T}}_L)$ and right $(\bar{\mathbf{R}}_R, \bar{\mathbf{T}}_R)$ camera coordinate systems respectively. Since the rotation matrices are orthogonal [71], $\bar{R}_L^{-1}$ can be written as $\bar{R}_L^T$. Comparing eq. 5.2 and eq. 5.6, we have,

$$\bar{\mathbf{R}}_{LR} = \bar{\mathbf{R}}_R \bar{\mathbf{R}}_L^T \qquad \text{, and} \qquad \bar{\mathbf{T}}_{LR} = \bar{\mathbf{R}}_R \left(\bar{\mathbf{T}}_L - \bar{\mathbf{T}}_R \right) \tag{5.7}$$

The transformation of a point from the right camera system to the left $(\bar{\mathbf{R}}_{RL}, \bar{\mathbf{T}}_{RL})$ can be derived similarly.

The close and far points in the left camera coordinate system, P_c^L and P_f^L, can be transformed to P_c^R and P_f^R in the right camera coordinate system using eq. 5.2. These can then be projected onto the right image plane as follows.

Collinearity Condition

After the computation of P_c^R and P_f^R, their projection onto the right image plane uses the collinearity condition. The projection is the reverse process the application of the range constraint (eq. 5.1, pg. 77). Consider, for example, the projection of $P_c^R(X_R, Y_R, Z_R)$ in right camera frame of reference to point $c(x_r, y_r)$ on the right image plane, we have (eq. 5.1)

$$-\frac{x_r}{f_R} = \frac{X_R}{Z_R} \quad \text{and} \quad -\frac{y_r}{f_R} = \frac{Y_R}{Z_R},$$
$$c(x_r, y_r) \quad = \quad (-f_r.\frac{X_R}{Z_R}, -f_r.\frac{Y_R}{Z_R}) \tag{5.8}$$

The projection of P_f^R to point 'f' on the right image is obtained in the same way.

Epipolar Constraint

Line segment c-f (fig. 5.2) is the epipolar segment obtained in the right image plane corresponding to $P_L(x_l, y_l)$ on the left image. Theoretically, the point in the right image corresponding to left image point, P_L, will lie *on* its epipolar segment in the right image plane. However, when real data and manual calibration (as compared to simulations) were used, this was not true because of calibration model limitations, sensor (image plane) discretization and random errors. Hence, the epipolar constraint was applied by admitting all right image point features 'close' to segment c-f (eq. 5.9) into the initial candidate set, CS_{P_L}, for P_L.

$$P_R^i \in CS_{P_L} \qquad \text{if } P_R^i \perp (c-f)_{P_L} \leq \tau \tag{5.9}$$

where, $P_R^i \in FS_R$ is the i*th* point-feature of the right image feature set and $CS_{P_L}, (c-f)_{P_L}$ are the candidate feature set and epipolar line segments corresponding to P_L of the left image; $\perp$ is symbolic for the perpendicular distance of the point (P_R^i) from the line segment and τ is the threshold for this distance.

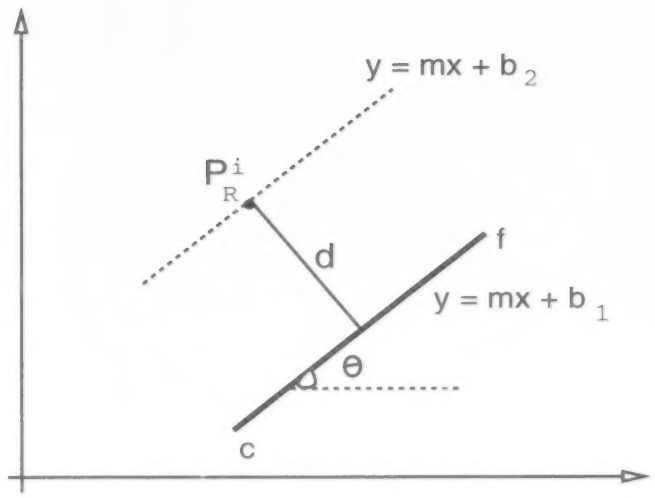

Figure 5.4: *Perpendicular distance (d) of feature from the epipolar line segment (c-f) used to decide inclusion into the candidate set.*

For the i*th* feature in the right image (P_R^i), the perpendicular distance (d_i) is calculated using the line in the right image plane parallel to the epipolar line segment (eq. 5.10, fig. 5.4).

$$d_i = |b_2 - b_1| \cos\theta \tag{5.10}$$

5.4 Refinement of Candidate Sets

Most registration methods apply constraints using surface smoothness, disparity gradient, cyclopean bounds, rigidity, and continuity to evaluate a feature's candidates for registration [6, 30, 42, 52, 60]. This process is terminated by the uniqueness condition, i.e. iterative evaluation of candidates ceases when one 'best' candidate is determined or all candidates have been discarded. For the evaluation of candidates, Matthies [42, 43] used coarse to fine correlation on image pyramids of different resolutions for both stereo-images. Others [6, 30, 52] only used correlation to get initial match probabilities which are iteratively improved by relaxation labelling using properties such as disparity gradient, figural continuity, surface smoothness, neighbour reinforcement, etc. to converge to a unique match or none at all. These approaches are aimed at

realizing a quick, sparse model of the 3D environment surrounding a robot, for instance. The cameras for these purposes are configured in a near-parallel geometry with baselines rarely exceeding 300mm. Tests in the operating room involved baselines varying between 300 mm and 500 mm. Larger baselines were used for greater accuracies. Large baselines, coupled with the fact that ball-features do not lie on any particular smooth surface, rule out the efficient use of the above properties *for the application of uniqueness* for convergence of initial candidate sets to singular or null sets. They could, however, be used for the purpose of refinement of candidate sets. Refinement of candidate sets, the intermediate stage in the correspondence hierarchy, aims to reduce the size of candidate sets by eliminating candidates which did not satisfy the constraint which was enforced at this level.

Experiments were conducted to test the feasibility of using correlation-based measures to reject candidate locations whose neighbourhoods bore little structural similarity to the feature's neighbourhood. Two camera configurations were used and stereo-images acquired using both configurations were tested. One configuration was defined as having a small baseline (320 mm) and vergence (24°) and the other as having a large baseline (420 mm) and vergence (32°). There were 12 ball-features visible in the stereo-images of both configurations. Square windows (side 75 pixels) with features in their center were selected. These were called feature subimages. Cross-correlation results were recorded for correlation between feature subimages in one image with feature subimages in the other image of the stereo-pair. Correlation used eq. 3.4 also applied during feature matching.

Tables 5.1 and 5.2 list the results of cross-correlation of the 12 feature subimages (size 75x75 pixels) of the left image with the 12 features of the right image for stereo-images of the two configurations respectively. For each feature subimage of the left stereo-image, the maximum and minimum correlation values (ρ) are given with the correlation value for the matching subimage in the right image. Often, the matching subimage will not have the maximum correlation value due to the perspective differences between the two views. The percentage of the ratio of this value with the maximum is indicative of the difference in likelihood between the matching subimage

Small Baseline-Vergence Configuration					
Feature Number	Cross-correlation Values, $\rho \in [-1, 1]$				
	ρ_{max}	ρ_{avg}	ρ_{min}	ρ_{match}	$100\frac{\rho_{match}}{\rho_{max}}(\%)$
#0	0.705	0.066	-0.359	0.705	100.0
#1	0.706	0.139	-0.220	0.706	100.0
#2	0.473	0.057	-0.173	0.473	100.0
#3	0.535	0.205	-0.171	0.535	100.0
#4	0.738	0.229	-0.147	0.738	100.0
#5	0.408	0.122	-0.159	0.408	100.0
#6	0.286	0.077	-0.065	0.286	100.0
#7	0.674	0.252	-0.100	0.674	100.0
#8	0.252	0.058	-0.130	0.226	89.6
#9	0.129	0.008	-0.084	0.129	100.0
#10	0.503	0.138	-0.313	0.503	100.0
#11	0.425	0.008	-0.501	0.425	100.0

Table 5.1: *Structural Similarity with a small baseline-vergence configuration (Baseline = 320mm, Vergence = 24°). Only feature #8 did not register the maximum similarity value with its corresponding point in the right image.*

Large Baseline-Vergence Configuration					
Feature Number	Cross-correlation Values, $\rho \in [-1, 1]$				
	ρ_{max}	ρ_{avg}	ρ_{min}	ρ_{match}	$100\frac{\rho_{match}}{\rho_{max}}(\%)$
#0	0.212	0.078	-0.046	0.212	100.0
#1	0.252	0.005	-0.201	0.252	100.0
#2	0.342	-0.034	-0.322	0.342	100.0
#3	0.356	0.081	-0.185	0.236	66.6
#4	0.221	0.037	-0.185	0.199	90.0
#5	0.341	0.033	-0.224	0.191	55.1
#6	0.297	0.084	-0.061	0.231	77.7
#7	0.397	0.012	-0.286	0.397	100.0
#8	0.265	0.043	-0.204	0.265	100.0
#9	0.307	0.111	-0.072	0.307	100.0
#10	0.220	0.002	-0.182	0.220	100.0
#11	0.324	0.027	-0.205	0.324	100.0

Table 5.2: *Structural Similarity with a large baseline-vergence configuration (Baseline = 420mm, Vergence = 32°). Feature #3, #4,#5, and #5, did not register the maximum similarity value with their corresponding points in the right image. There are many more features that do not register maximum similarity with their matching candidates as compared to table 5.1. This indicates the difference in perspective views as the baseline is increased.*

and the most structurally close subimage in the right image to the feature subimage in question from the left image. For the small baseline-vergence configuration (table 5.1), all features except #8 recorded maximum correlation values with their corresponding subimages. The correlation value recorded between the subimage for feature #8 and its corresponding subimage was 89.6% of the maximum correlation value registered for all subimages with feature #8. With the small baseline-vergence configuration, a threshold of 80% of the maximum correlation value was found to always include the matching feature in the refined candidate set. With the large baseline-vergence configuration (table 5.2), features #3 (66.6%), #4 (90.0%), #5 (55.1%), and #6 (77.7%) did not register maximum correlation values with their matching candidates. The threshold value for this configuration, based on testing with more stereo-images, was set at 50%. This is significantly lower than the threshold of 80% for smaller baseline-vergence configuration. The difference in thresholds was expected because the structural similarity between matching scenes deteriorates with increased baseline and vergence.

In the correspondence algorithm implemented, candidate sets were optionally subjected to the structural constraint as the intermediate stage of correspondence. A given feature's subimage was tested with large neighbourhoods of its candidates (members of its candidate set) in the other image of the stereo-pair. Refined candidate sets only admitted candidates with similarity values above the threshold relative to the candidate with maximum similarity (eq. 5.11). This does not ensure that the refined candidate set will be singular.

Given a feature L_i with a candidate set, $\mathcal{C}^i = \{r_1, r_2, \ldots, r_j, \ldots\}$. The refined candidate set ($\mathcal{R}^i$) for L_i is defined as:

$$\mathcal{R}^i = \left\{ r_k \ni r_k \in \mathcal{C}^i, \text{ and } r_k > \tau \left(\max_{r_j} \rho_{ij} \right) \right\} \tag{5.11}$$

where τ is the threshold chosen for the particular camera configuration, and ρ_{ij} denotes the cross-correlation coefficient between the feature subimages corresponding to L_i and r_j.

5.5 3D Feature Reconstruction

This section deals with the use of stereo-camera parameters for the reconstruction of possible ball-features in three dimensional space. As no uniqueness criterion was enforced during the refinement of candidate sets, there may be multiple members from the other image in a feature's refined candidate set. Thus, reconstruction with these may yield multiple 3D points for a single feature.

In photogrammetry, the process of reconstruction in 3D using two or more image point locations is termed space intersection. The concept of space intersection is presented first, followed by the model used for the calculation of three dimensional features.

In fig. 5.5, a vector relationship can be formulated (eq. 5.12) between the baseline vector, $\mathbf{B}$, left image vector, $\mathbf{a}_L$, residual parallax vector, $\mathbf{P}_1\mathbf{P}_2$, and right image vector, $\mathbf{a}_R$. If $[p_x, p_y, p_z]^T$ are the components of a vector in the direction of $\mathbf{P}_1\mathbf{P}_2$, by solving for the scalars $(\lambda_1, \lambda_2, \kappa)$(eq. 5.12), we can determine $\mathbf{P}$(X,Y,Z), corresponding to the object-space vector $\mathbf{WP}$.

$$\mathbf{B} = \lambda_1 \mathbf{a}_L + \mathbf{P}_1\mathbf{P}_2 - \lambda_2 \mathbf{a}_R \tag{5.12}$$

which in object-space is

$$\begin{bmatrix} B_x \\ B_y \\ B_z \end{bmatrix} = \lambda_1 \mathbf{M}_L^T \begin{bmatrix} x_l \\ y_l \\ -f_L \end{bmatrix} + \kappa \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} - \lambda_2 \mathbf{M}_R^T \begin{bmatrix} x_r \\ y_r \\ -f_R \end{bmatrix}$$

$$\begin{bmatrix} \lambda_1 \\ \kappa \\ \lambda_2 \end{bmatrix} = \begin{bmatrix} u_l & p_x & -u_r \\ v_l & p_y & -v_r \\ w_l & p_z & -w_r \end{bmatrix}^{-1} \begin{bmatrix} B_x \\ B_y \\ B_z \end{bmatrix} \tag{5.13}$$

where,

$$\begin{bmatrix} u_l \\ v_l \\ w_l \end{bmatrix} = \mathbf{M}_L^T \begin{bmatrix} x_l \\ y_l \\ -f_L \end{bmatrix}, \quad \begin{bmatrix} u_r \\ v_r \\ w_r \end{bmatrix} = \mathbf{M}_R^T \begin{bmatrix} x_r \\ y_r \\ -f_R \end{bmatrix},$$

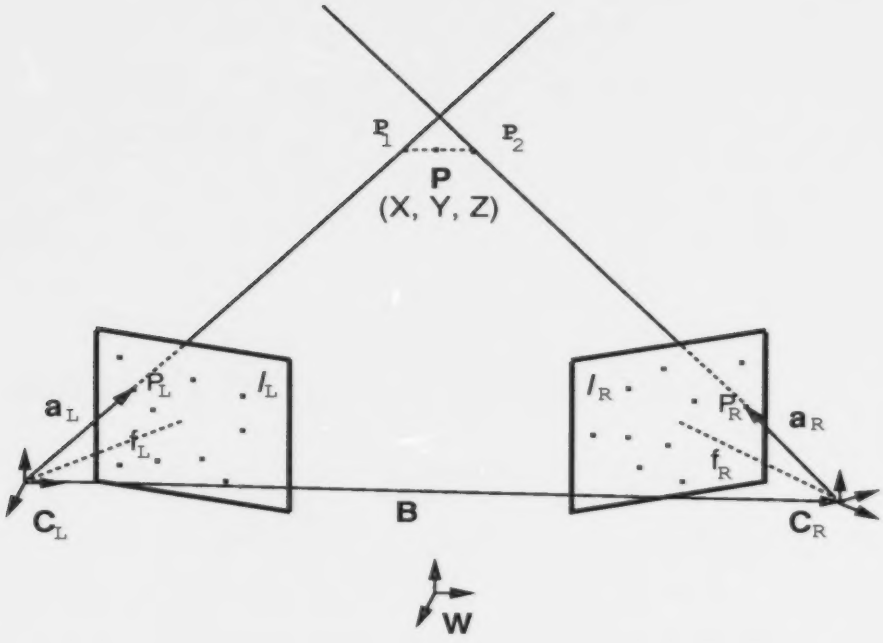

Figure 5.5: *Space Intersection: The position of* $\mathbf{P}(X,Y,Z)$ *is given by the vector relationship between the baseline vector, lines of sight, and the perpendicular* ($\mathbf{P}_1\mathbf{P}_2$) *between the LOS.*

$$\mathbf{B} = \mathbf{C}_L\mathbf{C}_R, \text{ and } \mathbf{P}_1\mathbf{P}_2 = \kappa \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} = \kappa\,(\mathbf{a}_L \mathbf{x} \mathbf{a}_R) \tag{5.14}$$

using eq. 5.13, we have,

$$\mathbf{WP} = \mathbf{WC}_L + \lambda_1 \mathbf{a}_L + \frac{1}{2}\mathbf{P}_1\mathbf{P}_2 \tag{5.15}$$

The above approach places the object-space point, $\mathbf{P}$, at the midpoint of the residual parallax vector, $\mathbf{P}_1\mathbf{P}_2$. This method for the reconstruction of $\mathbf{P}$ is somewhat *ad hoc* and lacks accuracy [46]; it has been presented as one easy approach to understand the process of 3D reconstruction from.

A C routine (CSPACO) was written based on a FORTRAN implementation [50] (SPACO) of the iterative reconstruction based on collinearity (eq. 3.4). The iterative

approach has higher accuracy, and can accommodate multiple lines of sight (LOS) corresponding to multiple cameras [70]. Details of the formulation of condition equations and their iterative solution are not relevant to the process of correspondence, and have not been covered here.

5.6 Triad-marker Identification

Most registration methods seek to find a unique match from a feature's initial set of match-candidates. To do this, a uniqueness constraint is typically propagated until a single candidate remains otherwise, the feature is discarded unpaired [6]. With OR images, however, there were no efficient constraints available at the stage of refinement to apply the uniqueness condition. Hence, a one-many correspondence was allowed. This meant that refined candidate sets could contain more than one candidate. The uniqueness constraint was not applied until after reconstruction when object-space was tested for a known number of triad-markers. An unoccluded triad-marker yields three 3D point features after detection, correspondence, and reconstruction. As the spatial configuration of these features was known from the geometry of the triad-marker (fig. 4.1), object-space was searched for triplets of points which satisfied the marker model. Heuristic constraints were used to eliminate false triads. This section describes the identification of triad-markers in 3D - the final level in the correspondence hierarchy (fig. 5.1).

3D Model Matching

The 3D model matching problem may be stated thus: given a set of 3D point features, a 3D model of the distances and angles between the three components of a triad-marker, and knowledge of the number of triads in object-space, identify the triads and their component points as mutually exclusive triplets from the set of 3D points. Generalized solutions for point structure matching use correspondence between frames [2] to arrive at camera and/or object motion parameters. Mitiche and Bouthemy [45] have presented a rigorous solution for generalized correspondence of 4-point structures for solving of motion parameters.

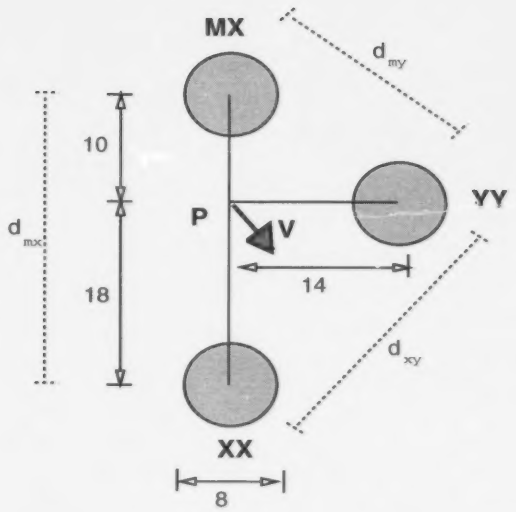

Figure 5.6: *A flat, planar triad model (all dimensions in mm).*

To monitor spinal surgeries, markers were designed as triads of spherical balls conforming to a model (fig. 5.6). The ball-components were 8 mm in diameter and together with their 'legs' (10mm, 14 mm, 18 mm) were configured in a plane. Previously illustrated markers may depict different configurations or lengths of legs. However, it was decided to use this configuration as it presents minimal occlusion problems. There are three distances possible between the three points that constitute a triad (fig. 5.6). These were used to define distance matrices for possible triad configurations.

Distance Matrices (DM) are defined as 3x3 upper triangular matrices consisting of 3D inter-feature distances for a prospective configuration being matched to the ideal triad model. A reference distance matrix was calculated for the model (table 5.3). For a triplet of points, if the DM or any permutation on the DM was 'similar' to the reference distance matrix, a possible triad was labelled (eq. 5.16). The triad error, used to quantify similarity to the reference matrix, was the Euclidean distance between the set of three inter-feature distances for any prospective triad with the corresponding set of inter-feature distances of the ideal triad model (table 5.3).

Given a subset , $\mathbf{T} = \{P_1, P_2, P_3\}$, of the 3D feature set, $\mathbf{T}$ is labelled a triad if,

Reference DM			Typical DM			Error Matrix		
XX	YY	MX	P_1	P_2	P_3	P_1	P_2	P_3
0.0	22.8	28.0	0.0	22.6	28.1	0.0	0.2	0.1
	0.0	17.2		0.0	17.4		0.0	0.2
		0.0			0.0			0.0
Matrix Error, $\varepsilon = \sqrt{(0.2)^2 + (0.1)^2 + (0.2)^2} = 0.3$								

Table 5.3: *The reference distance matrix (DM), a typical DM with associated error matrix (all dimensions in mm).*

for this permutation of point positions, distance errors (d_{12}, d_{23}, d_{13}) are such that:

$$\begin{aligned} |d_{12} - d_{my}| &< \tau \text{ , and} \\ |d_{23} - d_{xy}| &< \tau \text{ , and} \\ |d_{13} - d_{mx}| &< \tau \end{aligned} \tag{5.16}$$

where, d_{ij} represents the distance between P_i and P_j, τ is the tolerance of a 3D distance from the model distance, and (d_{my}, d_{xy}, d_{mx}) are the model distances (fig. 5.6). Distance matrix error (ε) for a configuration $\mathbf{T}$ is defined as:

$$\varepsilon = \sqrt{(d_{12} - d_{my})^2 + (d_{23} - d_{xy})^2 + (d_{13} - d_{mx})^2} \tag{5.17}$$

For every set with three members, there were 6 permutations (3!) which corresponded to the different configurations of the points tested relative to the model components using eq. 5.16.

Heuristic Constraints

Though not necessary with simulation data, with real data and manual calibration of camera parameters, heuristic constraints were necessary to eliminate any 'false' triads which may have accidentally satisfied the 3D model matching criteria (eq. 5.16). These criteria allowed large tolerances so that any true triad does not get rejected if the reconstruction or detection of any of its components is poor. The heuristics, applied to triads, could be considered equivalent to the uniqueness properties most correspondence methods apply to image features.

These heuristics, along with their basis and a description of their application, are stated here. These were applied to triads, 3D point feature configurations, each of

which had satisfied the triad model matching criteria (eq. 5.16). A triad, or triad set, has 3 members, each an object-space point reconstructed using an image feature from both stereo-images. Each triad also has a corresponding DM and triad error.

- **[H1] The six image features used to reconstruct the 3 components of the triad must be distinct.**

  This heuristic applies uniqueness to image features used for the components of any triad. If either the three left image features or the three right image features are not distinct, the triad set can be discarded.

- **[H2] A 3D feature can only belong to a single triad.**

  This heuristic applies uniqueness to 3D points. If a 3D feature belonged to more than one triad, only the triad with minimum triad error was retained. Given triads T_i, $i \in Z$, containing the common point. Only triad T_r is retained where, if ε_i is the triad error (eq. 5.16) of T_i,

  $$\varepsilon_r = \min_i \varepsilon_i \tag{5.18}$$

- **[H3] A single image feature can be used to reconstruct an object-space point in no more than one triad.**

  This heuristic applies uniqueness to 2D image features as in [H1] but considers features from different triads rather than within each one. If multiple triads have component object points computed using the same image feature, the triad with minimum distance error is chosen (eq. 5.18).

- **[H4] Triads, when arranged in order, should conform to intervertebral measurements.**

  The spine is roughly aligned along the x- or y-axis. Hence, when triads are sorted and placed along the x- or y-axis, distances between triads should (roughly) correspond to the intervertebral distances of the vertebrae which have triads rigidly attached to them. Intervertebral distances were obtained from x-ray data for real cases, and by manual measurement for simulations. If these measurements are not available, this heuristic is ignored. This heuristic was not used and

is mentioned for future work. The correspondence process was successful in locating triads without this constraint.

- **[H5] When complete, the number of triads should be the same as the number of triad-markers used.**
  This heuristic specifies the uniqueness of the number of triads detected. If true, this condition signals the end of the correspondence process.

5.7 Triad Representation and Results

This section covers the calculation and representation of triad information followed by typical results using operating room data with different camera configurations.

Representation of Triads

For visualization and recording progress during surgery, three pieces of information are required for every triad-marker which is automatically located by the correspondence algorithm. These are its label (corresponding to the vertebra to which it is rigidly attached), its position, and its orientation in object-space. For each triad-marker, the position and orientation information determined must be consistent with the 3D model of the triad. Though many triad prototypes were designed and tested, the ones finally selected for use were arranged in a flat, planar configuration (fig. 5.6) parallel to the xy-plane of the object space coordinate system. The position of point P (fig. 5.6, 5.7), defined as representative of the triad position, can be calculated when the spatial locations of 3D features corresponding to model locations XX, YY, and MX are known. The three triad component points constitute a plane in 3D. The orientation corresponding to a triad-marker was defined as the normal to this plane.

The triad position was defined as the point (**P**) which intesects the 3D line joining model points MX and XX in the ratio 10:18 or 5:9 (fig. 5.7). The orientation was taken as the unit vector in the direction of the cross product between the vectors **A** (XX-MX) and **B** (XX-YY) (fig. 5.7).

The labelling of triads was possible using the relative positions of the triads along the x- or y-axis of object-space. Since calibration is usually done with the x- or y-axis

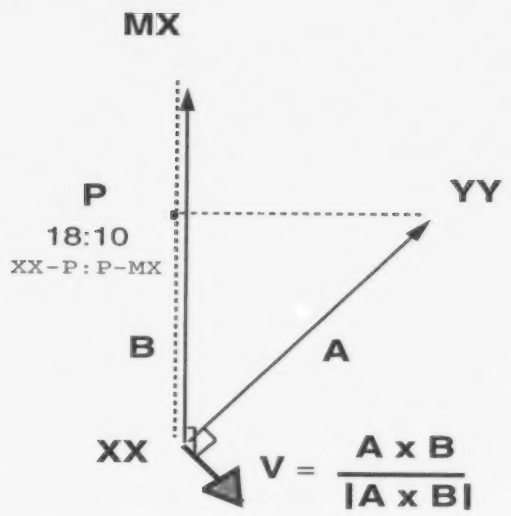

Figure 5.7: *Model-based calculation of triad position and orientation (all dimensions in mm).*

of the world coordinate system oriented along the OR bed (the patient's spine). There is *a priori* information regarding the vertebrae to which markers were attached, the triad-markers automatically identified in space could be labelled by the vertebrae to which they were attached by sorting them by the x- or y-coordinate of their position vectors.

Results

As the correspondence approach chosen was hierarchical, each stage was tested using simulations or test data followed by real data. Three dimensional points projected onto image planes during simulations required only the lowest level of the hierarchy for matching; i.e. the candidate sets of simulated point features were singleton sets as the matching point always occured *on* the epipolar line for the feature. With real data, however, the practical limitations of the calibration model, furthered by random error, become evident. Candidate sets contained multiple candidates. Often, the matching candidate was not the closest to the epipolar line segment of the feature. This reaffirmed the requirement of the intermediate stage of refinement to eliminate obvious mismatches based on different criteria before 3D reconstruction. Heuristics were designed to enforce uniqueness at the highest level of correspondence. As mul-

tiple 3D reconstructions with image points had been done by that stage, uniqueness was enforced for 2D image features, 3D object-space points, and triad-markers.

The integrated matching algorithm was tested using stereo-cameras in two different configurations, one with a small baseline and vergence and one with large baseline and vergence. For both configurations, the parameters (like threshold τ for candidate set refinement) used were the more relaxed values designed for the large baseline-vergence configuration. This was because parameters for the small baseline-vergence configuration were only devised as being optimal for speed, and not functionality.

For the camera configuration with small baseline-vergence, stereo-images for the final stage are shown in figs. 5.8, 5.9. The twelve feature locations (1, 2,... 11, 12) are marked along with added noisy points which formed the input to the correspondence algorithm. The correspondence process for the matching of a particular feature is now followed visually. Fig. 5.10 shows a feature in the left image. The lowest level of correspondence (fig. 5.1) yields its candidate set. The 8 members of its candidate set in the right image are shown in fig. 5.11. The intermediate level of refinement leads to the elimination of features { F, 5, 8, 9}, half of the candidate set. Fig 5.12 shows the remaining 4 members of its refined candidate set. After the application of uniqueness at the highest level (triad matching), the feature was found to belong to triad #2, and matched only a single feature in the right image (fig. 5.13).

As mentioned, there were two camera configurations, each used to record two-stage surgeries. There were four triads used in both surgeries. The 12 image features were labelled identically in the feature sets of the stereo-images (left image feature #i matches right image feature #i) so that the application of heuristics to discard 'false' triads can be easily understood. The correspondence algorithm returned the four triads configured correctly in both stages when both configurations were used. Results are provided for all triads configured in space (model matching), their triad errors, constituent model points, and image features. Tables 5.4, 5.5 show results for the initial and final stages respectively, when the small baseline-vergence configuration was used. Table 5.6 shows results of the initial and final stages when the large baseline-vergence configuration was used. The column labelled heuristics lists the heuristic

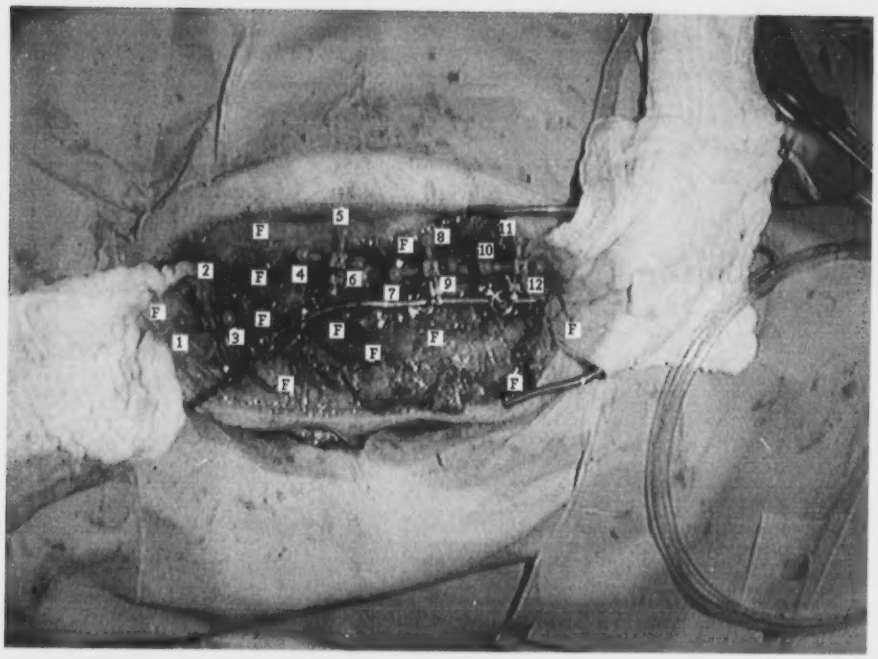

Figure 5.8: *Left stereo-image with annotated features (F for false positives).*

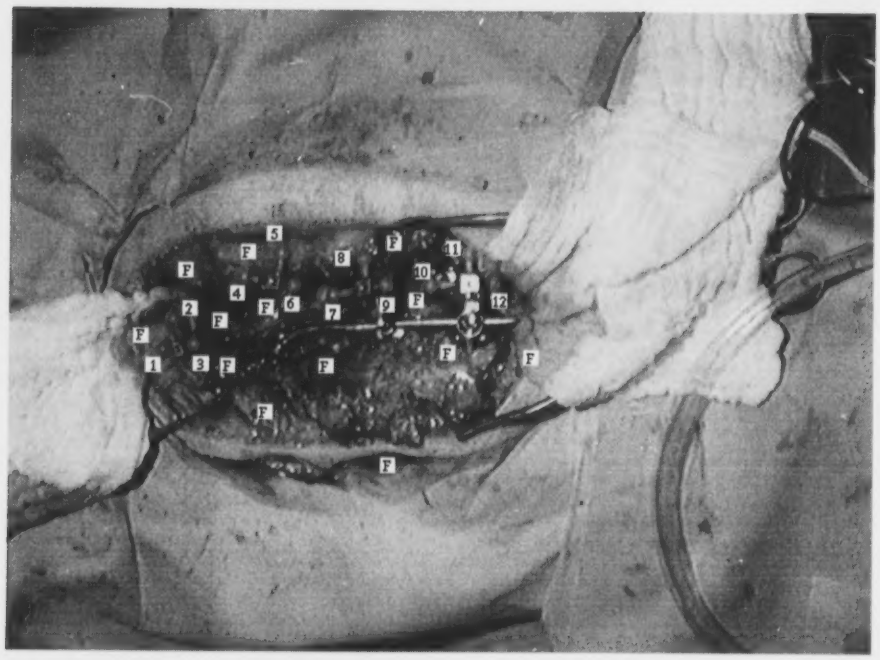

Figure 5.9: *Right stereo-image with annotated features (F for false positives).*

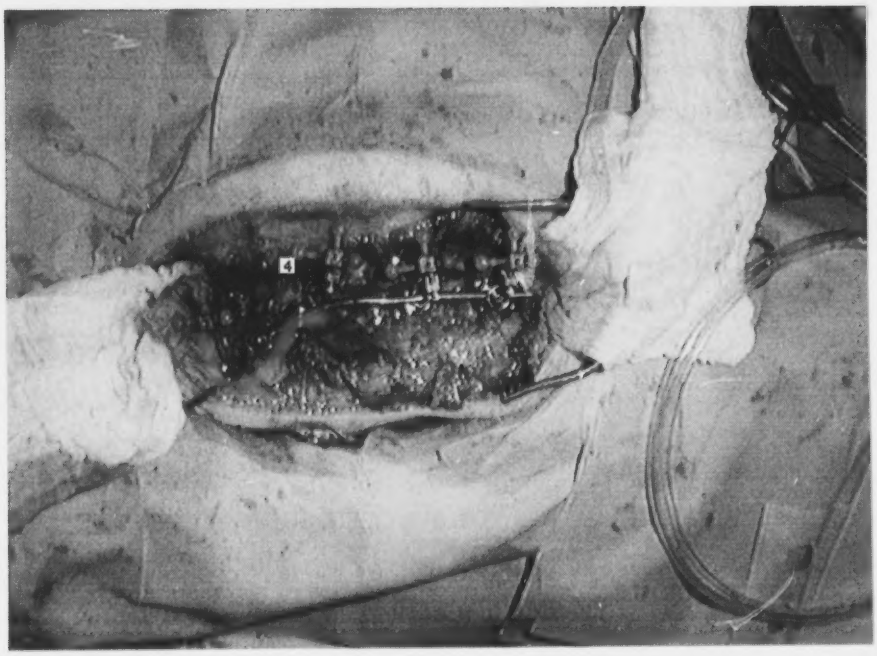

Figure 5.10: *Left image feature #4 (LIF#4) which is to be matched with a unique candidate in the right image.*

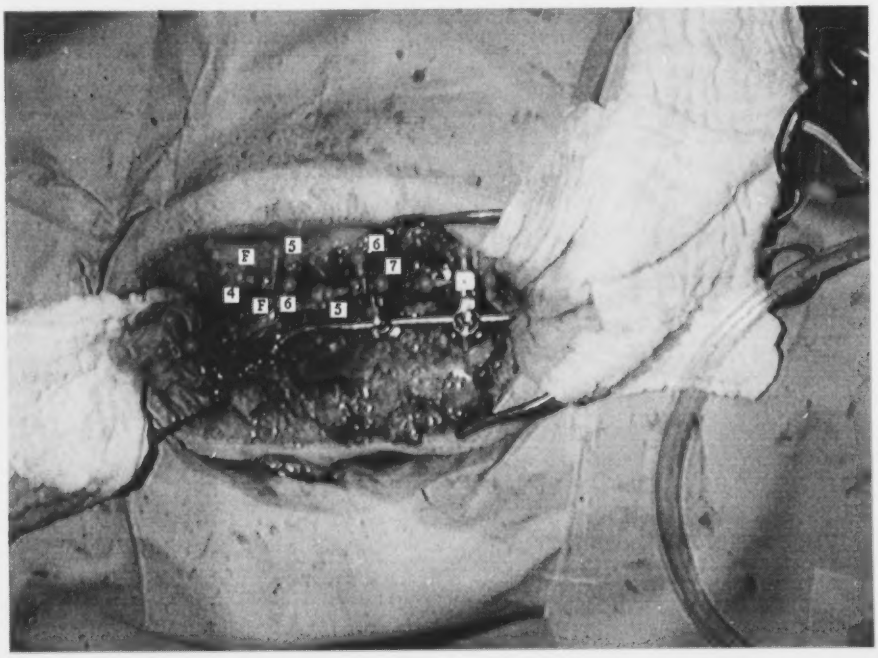

Figure 5.11: *Correspondence Stage I: Candidate set features in the right image for LIF#4.*

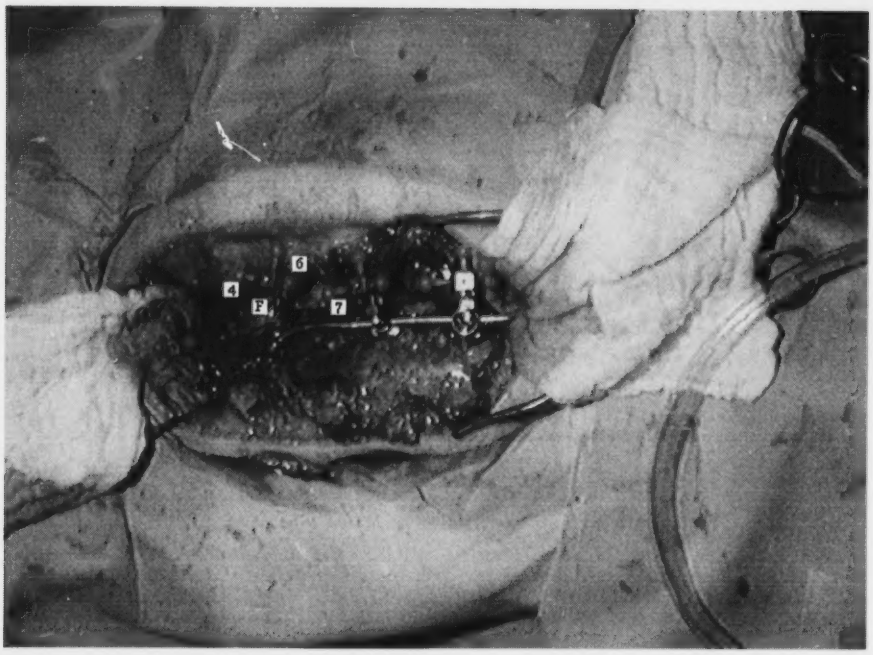

Figure 5.12: *Correspondence Stage II:The refined candidate set for LIF#4.*

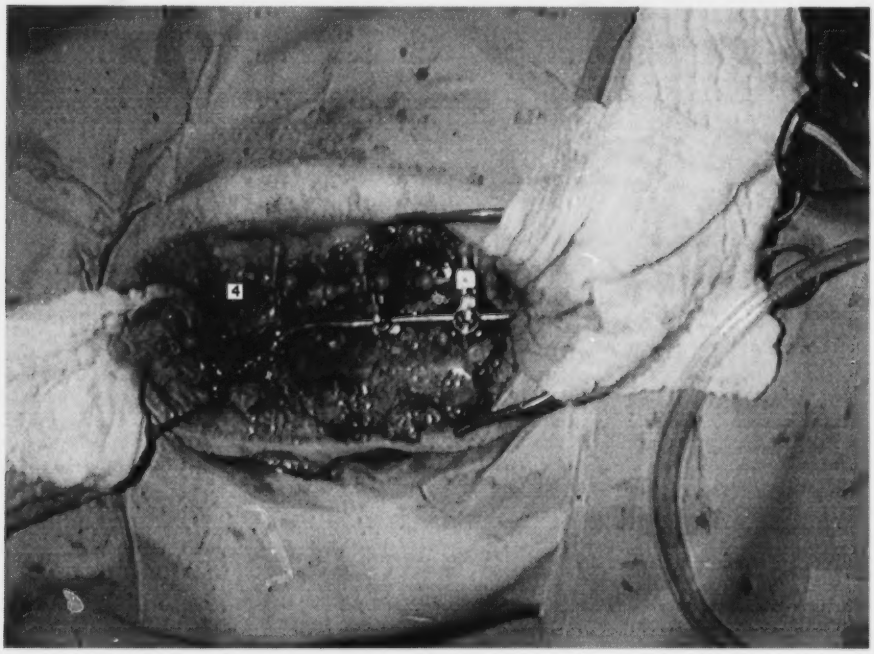

Figure 5.13: *Correspondence Stage III: The unique matching candidate for LIF#4 after triad identification.*

used to disqualify a particular triplet's configuration which satisfied the 3D model matching criteria. The model components (XX, YY, MX) are given in terms of the left and right image feature numbers used to construct the 3D component. Triad errors are also included so that the application of heuristics can be understood. In the column showing heuristics used to discard triads, 'P' indicates a triad which passed all heuristic evaluations. In each stage, all four triads present in the images were correctly located.

The correspondence process was successful using both camera configurations. This proved that the integrated paradigm used is robust enough to solve for baselines as large as 420 mm. The triad errors also prove that the accuracy of measurements made by the system on composite bodies (triads) is excellent. Simulations suggest that an improved calibration model (eg. one that takes sensor discretization into account) would obviate the need for the intermediate stage of correspondence presented here. The constraints designed here were dependent on the system design and prove that stereo-correspondence is a solvable problem in specific cases where *a priori* scene knowledge can be integrated with generalized constraints to provide a complete solution.

Small Baseline-Vergence Configuration (Stage 1)								
Triad #	Error (mm)	XX(L)	XX(R)	YY(L)	YY(R)	MX(L)	MX(R)	Heuristic
0	2.98	1	1	2	2	3	3	P
1	0.59	4	4	5	5	6	6	P
2	3.53	7	6	6	4	6	5	H1
3	1.17	9	8	9	9	8	8	H1
4	2.13	7	7	8	8	9	9	P
5	3.01	9	7	10	9	9	8	H1
6	3.59	12	8	11	8	10	7	H1
7	4.27	12	10	10	9	10	8	H1
8	1.37	10	10	11	11	12	12	P
9	4.70	10	7	11	8	12	8	H1
10	1.74	10	7	12	9	12	8	H1
11	5.65	10	7	11	8	12	9	H3

Table 5.4: *Triad matching with a small baseline-vergence configuration (Baseline = 320mm, Vergence = 24°). For this stage, model matching resulted in 12 possible triads. Triad errors are provided in mm. The components of the triad model (XX, YY, MX) are also given in terms of the left (L) and right (R) image features used to reconstruct the 3D point in each triad's component positions. The information in the table can be used to confirm the heuristic used to evaluate the triad marker (P implies that the triad passed all tests). Four triads were used in this surgery. They were all identified in the correct configuration.*

Small Baseline-Vergence Configuration (Stage 2)								
Triad #	Error (mm)	XX(L)	XX(R)	YY(L)	YY(R)	MX(L)	MX(R)	Heuristic
0	1.24	1	1	2	2	3	3	P
1	0.38	7	6	6	4	5	4	H1
2	0.50	4	4	5	5	6	6	P
3	4.06	8	9	9	9	8	8	H1
4	3.94	9	8	9	9	8	8	H1
5	4.25	10	6	9	4	8	4	H1
6	1.36	7	7	8	8	9	9	P
7	4.21	8	9	8	8	9	9	H1
8	3.78	9	8	8	8	9	9	H1
9	1.92	10	9	10	8	9	7	H1
10	3.99	11	6	10	6	9	4	H1
11	5.33	9	7	10	9	9	4	H1
12	4.84	8	9	9	9	10	10	H1
13	4.00	8	6	9	6	10	7	H1
14	4.37	12	8	11	8	10	7	H1
15	2.72	11	4	10	4	11	5	H1
16	3.58	11	4	12	6	11	5	H1
17	3.67	9	4	10	4	11	6	H1
18	3.93	10	6	12	7	11	6	H1
19	5.22	10	7	11	8	12	9	H2
20	3.25	12	7	11	6	12	6	H1
21	4.44	10	10	11	11	12	12	P
22	3.19	12	11	11	11	12	12	H1
23	4.29	12	6	11	4	12	4	H1
24	3.60	10	7	11	8	12	8	H1
25	2.55	10	7	12	9	12	8	H1

Table 5.5: *Triad matching with a small baseline-vergence configuration (contd.) (Baseline = 320mm, Vergence = 24°). Table notation as explained in table 5.4.*

Large Baseline-Vergence Configuration (Stage 1)								
Triad #	Error (mm)	XX(L)	XX(R)	YY(L)	YY(R)	MX(L)	MX(R)	Heuristic
0	1.22	3	3	2	2	1	1	P
1	0.51	3	3	2	2	1	2	H1
2	2.71	5	3	5	2	4	1	H1
3	0.87	6	6	5	5	4	4	P
4	0.74	9	9	8	8	7	7	P
5	2.33	8	8	9	9	9	8	H1
6	0.77	12	12	11	11	10	10	P
7	3.96	12	7	10	7	12	9	H1
Large Baseline-Vergence Configuration (Stage 2)								
Triad #	Error (mm)	XX(L)	XX(R)	YY(L)	YY(R)	MX(L)	MX(R)	Heuristic
0	2.06	3	3	2	2	1	1	P
1	4.21	6	3	5	2	4	1	H3
2	0.61	6	6	5	5	4	4	P
3	2.71	6	6	5	5	4	5	H1
4	3.14	9	4	9	6	7	4	H1
5	2.67	7	8	8	8	7	7	H1
6	0.56	9	9	8	8	7	7	P
7	3.01	10	1	10	3	9	2	H1
8	1.00	9	4	10	6	10	5	H1
9	3.21	9	5	10	6	10	5	H1
10	3.03	11	12	11	11	10	10	H1
11	0.76	12	12	11	11	10	10	P
12	1.78	11	10	10	10	11	11	H1

Table 5.6: *Triad matching with a large baseline-vergence configuration (Baseline = 420mm, Vergence = 32°). Table notation as explained in table 5.4.*

6 Visualization

This chapter deals with the meaningful representation of triad-marker data (position and orientation) automatically computed by the system. The purpose will be real-time, dynamic visualization of the spine during surgery, post-operative clinical or surgical evaluation, and the creation of a record of the different stages of surgery. Issues dealing with the interpretation of marker information in terms of vertebral movements are discussed to lay the foundation for future work. The background for a graphics visualization software [14, 63] is given. This software has the capability of representing 3D spine information as a set of vectors or vertebral-model objects. Triad information, automatically generated from previous algorithms, was graphically represented by this software using both vectors and graphical models of the spine.

6.1 Clinical Requirements

In the operating room, acquisition of stereo-images is followed by image processing, feature detection, and correspondence. These processes return position and orientation of triad-markers in the form of 3D vectors corresponding to triad-markers. These vectors, called triad vectors, exist in the object-space coordinate system defined during camera calibration. As the stereo-rig used was mounted on a mobile platform, triad vector sets corresponding to any stage of surgery, necessarily used one of the triad vectors as a reference. This was the triad vector corresponding to a reference triad which was placed on the vertebra expected to undergo minimal change during surgery, usually a lumbar or cervical vertebra. Triad parameters between different stages could be compared using position values relative to those of the reference.

Clinical studies and visualization require knowledge about the 3D status of vertebrae or the spine during different stages of surgery. The camera system measures position and orientation changes in the triad vectors. These cannot be directly interpreted as changes in vertebral 'body' vectors, virtually defined 3D vectors which represent a vertebra's position and orientation. The body vector could be taken as a vector positioned on the spinal midline which passes through the vertebra's center of

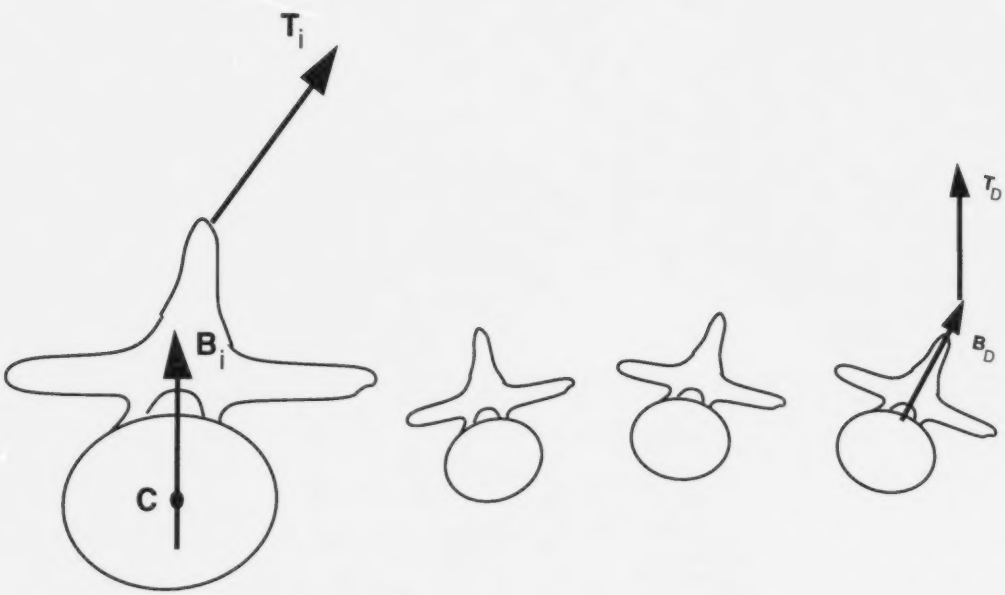

Figure 6.1: *Body and triad vectors for the* ith *vertebra* ($\mathbf{B}_i$,$\mathbf{T}_i$). *The vectors are also shown for the reference (dummy) vertebra* ($\mathbf{B}_D$,$\mathbf{T}_D$).

mass. In this work, body vectors are implicitly defined in graphical routines where vertebral-objects are modelled. The body ($\mathbf{B}_i$) and triad ($\mathbf{T}_i$) vectors of a vertebra are shown in fig. 6.1. As the surgeon may fix triad-markers at any location on the spinous process of a vertebra, there is no *a priori* knowledge regarding the 3D relationship between the triad and body vectors for a vertebra. The direct use of triad vectors for the representation of the spine during surgery would be inaccurate and difficult to visually interpret for the surgeon. Another related issue is that by comparing vectors (relative to reference vectors) between different stages, the 3D change in position and orientation of triad vectors can be inferred. Even if this were to be directly applied, as a gross approximation, to vertebral body vectors, the 3D status of the spine cannot be given without information regarding the initial (pre-operative) configuration of the spine.

The solution proposed was the use of stereo-radiographs of prostrate patients to calculate the preoperative 3D configuration of the spine. If the preoperative status of the spine is modelled, position and orientation changes recorded intraoperatively

using stereo-cameras can be used to update the configuration of the spine-model. The following sections present the current implementation of this approach and the issues yet to be dealt with during future work.

6.2 Graphical Representation

Based on a study of methods for measurements of vertebral position and rotation [56, 63] using radiographs, visualization software (SPINE) was developed [63] for the 3D modelling of the spine using IBM's GL (Graphics Language) graphics package. It was proposed to modify the existing SPINE program for the graphical display of results from stereo-cameras. Chu [14] ported SPINE to the OpenGL standard and made modifications to make it compatible with stereo-camera data.

The basis for the modifications made to SPINE were as follows. For any surgery, stage 0 represented the preoperative stage when instrumentation had been attached to the spine, but no active correction had been initiated. For this stage, lying stereo-radiographs had been analyzed [56] and the configuration of the spine (body vectors) was made available. This information was used by SPINE for the display of the uncorrected spine. For consequent stages, triad vector data, automatically computed by the stereo-camera system, was made available. For these stages, the triad vector data was be used to compute body vector information as follows. Consider stage 0 and stage 1 for the ith vertebra (fig. 6.1). When the system provides triad vector data for stage 1, 3D vector information is available for the following: $\mathbf{T}_i^0$ (triad vector, stage 0), $\mathbf{T}_i^1$ (triad vector, stage 1), and $\mathbf{B}_i^0$ (body vector, stage 0). It is required to compute $\mathbf{B}_i^1$, the body vector for the ith vertebra for stage 1. Rotation for a vector in 3D can be completely defined by an axis ($\mathbf{A}$) and an angle of rotation (θ) about that axis. The axis and angle are calculated using the cross product between the pre-correction and post-correction triad vectors (eq. 6.1).

$$\mathbf{A} = \mathbf{T}_i^0 \mathbf{x} \mathbf{T}_i^1$$

$$\theta = \sin^{-1}\left[\frac{|\mathbf{T}_i^0 \mathbf{x} \mathbf{T}_i^1|}{|\mathbf{T}_i^0|.|\mathbf{T}_i^1|}\right] \tag{6.1}$$

This 3D rotation ($\mathbf{A}$, θ) can be applied to $\mathbf{B}_i^0$ to arrive at $\mathbf{B}_i^1$. The positional difference between $\mathbf{B}_i^0$ and $\mathbf{T}_i^0$ can be maintained to arrive at the new position of $\mathbf{B}_i^1$. Post-correction body vectors are obtained in this fashion and displayed by SPINE. The underlying assumption is that body and triad vectors for the same vertebrae constitute rigid bodies and so, are subject to the same 3D rotation. Thus, SPINE attempts to represent 3D triads movements as measured by stereo-cameras in terms of changes to the configuration of the spine beginning from its precorrection configuration. Visually, this is more meaningful than displaying triad vector information only.

Images of a two-stage surgery were digitized. The feature sets were used with the correspondence algorithms described earlier. All four triad-markers were successfully located in both stages. Figures 6.2 and 6.3 show the results of reconstruction of the spine model using body vectors and graphical vertebral objects respectively. As lying radiographs of this particular patient were not available, *standing* radiographs of *another* patient were used to provide initial body vector information. This is not an accurate representation, but the purpose, at this stage, was only to simulate the visualization procedure. Triad vectors are transparent to visualization, but form composite objects with body vectors and are used to manipulate them in 3D. Triad vector data corresponding to the four triad-markers is given in table 6.1. It has been suggested [21] that a protocol be established wherein lying stereo-radiographs will be taken preoperatively in the clinic, before and after instrumentation, during *characteristic stages* of instrumentation and postoperatively in the clinic.

This graphical approach is only preliminary and is yet to be completely evaluated. Some of the issues related to this interpretation of triad data which can be investigated as part of future work are now discussed.

6.3 Outstanding Issues

As triad vectors do not correspond directly to body vectors of vertebrae, a number of issues arise during the interpretation of triad data for clinical purposes. These are dealt with in view of two suggested approaches to interpreting triad data: using OpenGL, as currently implemented, and using photogrammetry to relate camera

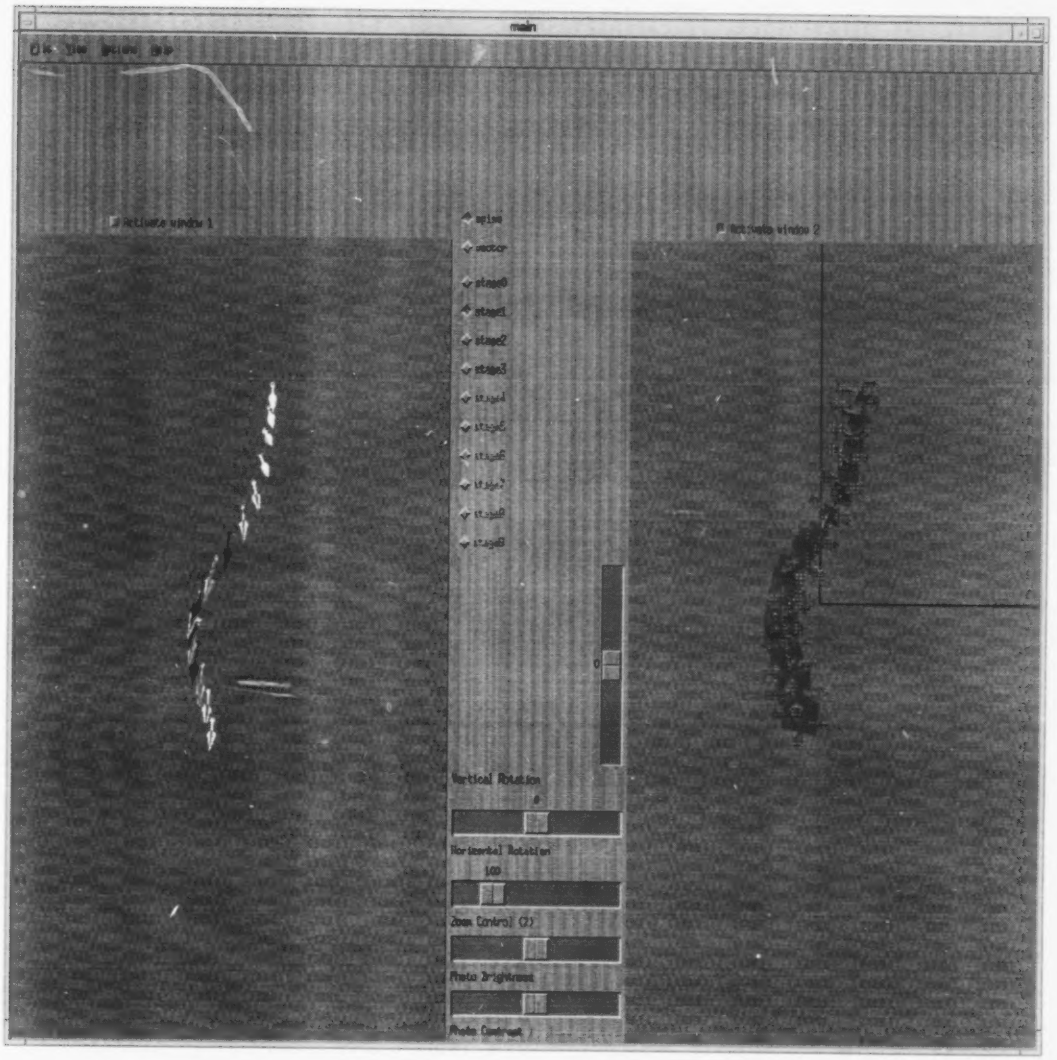

Figure 6.2: *Precorrection configuration of the spine using body vectors and vertebral models. The dark vectors corresponding to the vertebrae marked with triads. The remaining vectors are interpolated using the initial (precorrection) configuration of the spine.*

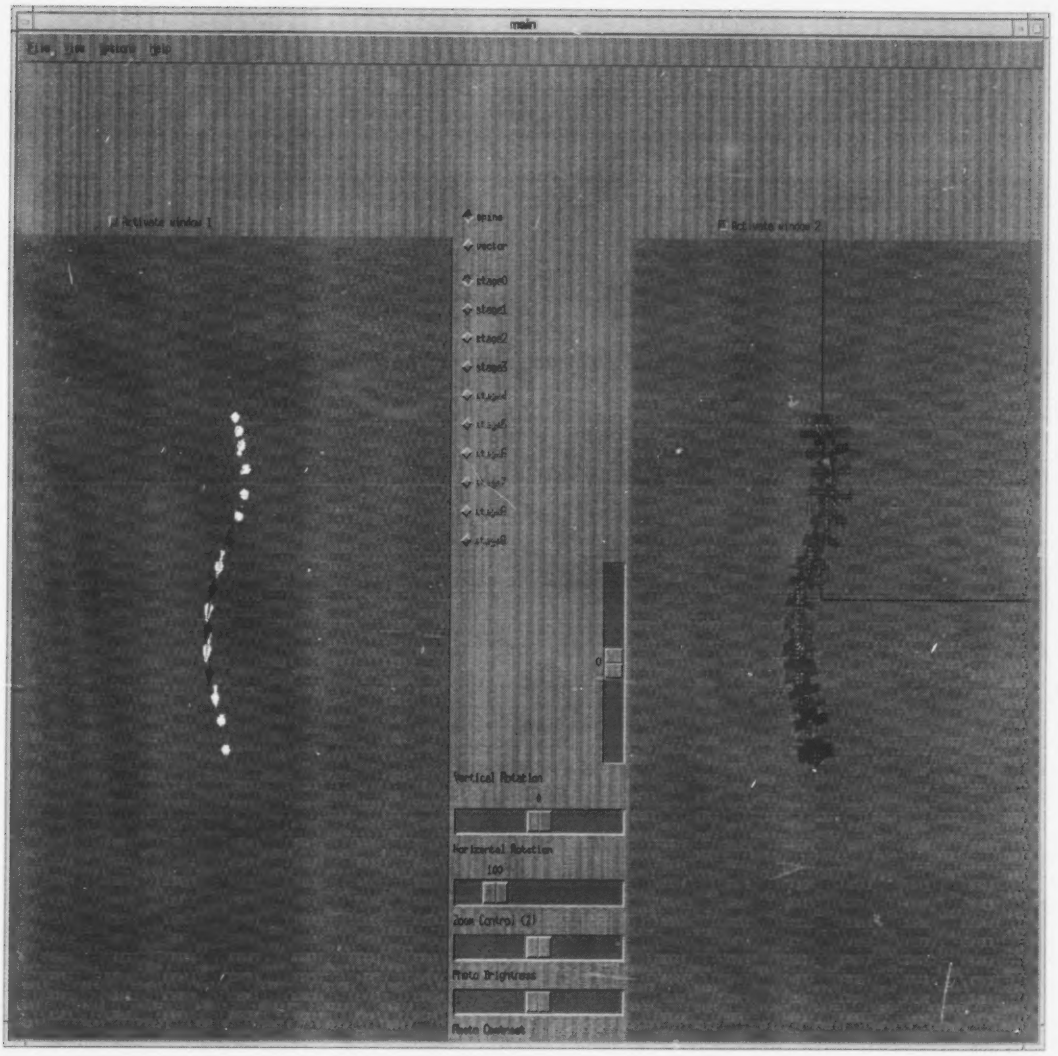

Figure 6.3: *Postcorrection configuration of the spine using body vectors and vertebral models.*

Triad Data (Pre- and Post- Stage) *(X,Y,Z in mm; α,β,γ in DDD.XX)*							
Stage	**Vertebra**	**X**	**Y**	**Z**	α	β	γ
Pre	L1	0.0	0.0	0.0	116.14	112.62	35.85
Post	L1	0.0	0.0	0.0	116.01	114.24	36.98
Pre	T11	26.18	-48.13	30.56	119.34	105.03	33.73
Post	T11	34.44	-59.67	29.32	79.08	99.65	14.65
Pre	T9	41.64	-97.76	40.59	86.66	111.70	22.01
Post	T9	33.32	-107.08	28.58	97.79	92.67	8.29
Pre	T7	51.50	-145.55	34.51	92.86	88.69	3.16
Post	T7	38.41	-156.18	17.53	86.58	74.78	15.57

Table 6.1: *Postion and orientation information for the 4 triads used in figs. 6.2 and 6.3. Pre- and post-correction data is grouped for each triad. (α,β,γ) are vector angles from the (X, Y, Z) coordinate axes respectively. The reference triad, planted on vertebra L1, shows little angular deviation due to correction.*

and radiograph object-spaces. Future work should concentrate on the evaluation of measurements recorded by the OpenGL approach for known 3D vertebral and triad vector motions. As an alternative, the latter approach is strongly recommended because of its mathematical rigour in establishing the relationship between camera and radiograph measurements. Using OpenGL is a more convenient approach for visualization rather than quantitative analysis of spine dynamics during surgery.

Relating camera and radiograph frames of reference

Currently, triad vector data, in the camera object space, and body vector data, in the radiograph object space, are combined in a common frame of reference in the 3D viewport provided by SPINE. Although both object-spaces represent real world measurements, it might prove more accurate to relate the two photogrammetrically, or at least do so in an experiment to prove that they are compatible. For any surgery, a common set of vertebral landmarks can be reconstructed using radiographs and the stereo-camera system. The seven-parameter transformation [46] can be applied to the two sets of data to arrive at the rotational and scale difference between measurements in the two frames of reference.

Interpretation of triad vector motion in terms of vertebral body vector

motion

As the motion of vertebrae is not well understood, it is difficult to relate positional and rotational changes in a triad vector to those of the body vector of the vertebra to which the triad is attached. A rigid translation of a vertebra in 3D will reflect identically on the triad. However, at this time, the trajectory of the triad vector has not been modelled in terms of rotation of the body vector about an arbitrary axis in 3D coupled with 3D translation. Experimental evaluation is required under controlled conditions to assess the validity of the currently implemented scheme of interpretation of triad motions using OpenGL.

The accurate interpretation of triad vectors in terms of body vectors is an important issue for the intraoperative monitoring of scoliosis surgery using this approach.

7 Conclusions and Recommendations

This chapter concludes the thesis in three parts. The first subsection briefly summarizes highlights of research done and results obtained. This work was intended to test the feasibility of using stereo-vision for intraoperative measurements. To build on this work, future directions of research and experimentation are outlined in the second section. Conclusions, based on the *intraoperative experience*, constitute the final section.

7.1 Summary of Results

Five distinct areas of work emerged when the stereo-vision paradigm [5] was applied for making intraoperative measurements. These were mechanical system design, stereo-camera calibration, feature detection in OR stereo-images, correspondence between feature sets returned by detection, model matching in 3D space, and visualization of data. Work done in each of these is summarized below.

Mechanical System Design A mobile platform was modified for the operating room. It accommodates a stereo-rig, a computer with frame-grabber, and a monitor. The stereo-rig consisted of an adjustable arm (3 degrees of freedom), a pair of Sanyo CCD cameras, and a variac-controlled light source. Many prototypes of triad-markers were designed with different fixation mechanisms so that they could be rigidly attached to the spinous processes of vertebra during surgery. Spherical ball-components of the triad-markers were fabricated using painted plastic. These ball-components were 8 mm in diameter and constructed with legs of variable length conforming to a specified triad model. All mechanical fabrication was done at the Electrical Engineering (EE) Workshop.

Stereo-Camera Calibration Calibration routines were developed for the Sanyo CCD sensors used. A control frame (*peg-board*) for calibration was designed guided by simulations. The control frame was constructed at the EE Workshop. Collinearity-based routines were used for calibration of stereo-cameras

and reconstruction of matched points in 3D. Accuracies of (0.3, 0.3, 0.7) mm were obtained in the (X,Y,Z) object-space directions respectively.

Feature Detection Routines were developed for the preprocessing of raw OR images using gamma correction, truncated median filtering, and Sobel edge detection. A novel algorithm, based on subimage statistics, was devised for the automatic recognition of the surgical field of view in OR images. This was used to reduce the size of images to save unnecessary computation during later stages. Gray level and edge-spaces were used for cross-correlation with multiple templates for the localization of ball-features. Six gray level and four edge-space template classes were developed by clustering and organized as template trees. Fundamental work was done to compare the characteristics of five different forms of cross-correlation. Cross-correlation with means adjustment and normalization (RHO), and means adjustment alone (CCM) performed well. DETECT POINT, the implementation of this approach, was tested on a three-stage surgery. All unoccluded ball-features were successfully located within one pixel from manually specified locations. The feature sets returned by DETECT POINT contained approximately one false positive for every true feature detected.

Correspondence A three-stage hierarchical approach was adopted for correspondence between feature sets returned by the system's detection algorithms. The lowest stage collects candidates for each feature from the other image of the stereo-pair based on epipolar and range constraints. The intermediate stage rejects candidates based on structural similarity between feature and candidate neighbourhoods. This refines candidate sets constructed by the first stage. Features are reconstructed in 3D with each of the members of their refined candidate sets. At the highest level, 3D model fitting was implemented to match triplet 3D point configurations to the 3D triad model. False triads were eliminated using uniqueness-based heuristic rules. Software developed for this correspondence model was tested with data for two camera configurations - small baseline

(320 mm) and large baseline (420 mm). All triads were successfully located in their correct configurations. All other points were rejected as false positives or incorrectly reconstructed 3D features.

Visualization Issues were presented regarding the interpretation of triad position and orientation information in terms of vertebral movements during stages of surgery. Specifications were designed for Chu [14] for the modification of the program (SPINE) used to display vertebral information. SPINE can be currently used to visually assess the configuration of the spine after any stage of surgery. An approach using photogrammetry to relate camera and radiograph coordinate systems was proposed for future work.

7.2 Recommendations for Future Work

If future work is to build on what has been done, the theory, related literature, software implementation, and limitations of work done must be *thoroughly* understood *before* proceeding. For this purpose, relevant theory has been referenced and explained and software written has been documented in detail. This section provides the author's recommendations for future work based on his experience with the project and its limitations. The five areas of work used to describe results in the previous section are also used here.

Mechanical System Design

1. When the development of the system is deemed complete, it should be possible to mount the stereo-rig (arm accommodating stereo-cameras and lamp) rigidly to the ceiling. The lenses can be replaced by more powerful zoom lenses (focal length 28-50 mm should be sufficient) to maintain the field of view. The lamp may need an optical accessory to avoid dispersion of light. As the field of view will only be slightly larger than that of the mobile system, there will be no need to replace the cameras or system software.

2. The aperture setting of the cameras is currently of an inconvenience during cali-

bration *and* intraoperative image capture. As the aperture settings are manually adjustable and have no f/stop locks, it is difficult to visually adjust the f/stop on both cameras so that their apertures are equally open. The Sanyo VDC-2524 manual suggests the use of automatic iris lenses (model VCL-X, where X refers to the desired focal length). These lenses feed back to the auto-iris circuitry already in the cameras to provide pictures with consistent lighting. This would eliminate the tedium of matching the lighting (apertures) of the left and right camera images so that detection procedures do not have problems detecting visible features. It is expected that because of standardized lighting, the auto-iris will not vary significantly to affect reconstruction accuracy.

3. To eliminate errors during image acquisition, it is advisable to devise a procedure which uses the EXT SYNC circuitry to take synchronized stereo-images in response to single *capture* signal. This prevents left and right images being taken during different phases of the patient's breathing cycle.

4. Triad-markers of different sizes can be constructed in the EE Machine Shop, if surgeons so desire.

Stereo-Camera Calibration

1. As suggested during the survey of calibration techniques conducted for this project, Tsai's [36, 62] two stage calibration method can be tested and compared to the classical collinearity method [51]. Tsai's method involves constraints that account for sensor discretization. Collinearity, having its roots in photogrammetry, works with continuous plate coordinates. Only experimental evaluation can prove whether Tsai's method does provide any *significant* increase in accuracy. Software for Tsai's method, as implemented by Willson is available (http://www.cs.cmu.edu/afs/cs.cmu.edu/user/rgw/www/TsaiCode.html). A comparison of Tsai's method with Wei and Ma's two plane calibration approach is already available [66]. The Calibrated Imaging Laboratory (CIL) of Carnegie Mellon University (http://www.cs.cmu.edu/ cil/cil.html) is a place

where related work on stereo-vision and image understanding is taking place. Researchers at the CIL can be contacted for advice on practical issues regarding stereo-vision as they arise.

2. After every calibration trial, data was collected on the accuracy of reconstruction with the calibrated camera geometry. Accuracies as good as 0.7 mm in the Z-direction were recorded. Experiments should be conducted to measure three dimensional angles using such camera geometries with sub-millimetre reconstruction accuracies. Statistical analysis of these measurements can be used to establish a relationship between measurements of distances and angles in three dimensions. These experiments need not employ markers, but may need a carefully constructed and measured control frame with a wide range of angular edges in 3D.

3. If FORTRAN routines like V2STEREO, NDLT, and SIM3D are to be used on a permanent basis, the programs should be ported to the C language (SPACO has already been ported). They should be formally documented with the relevant photogrammetry and mathematical theory needed to understand and modify the programs. Currently, calibration is time consuming because it requires manipulation of data files in different formats which are to be used by the general purpose calibration programs mentioned above. The ported C routines can be combined into an integrated, automatic calibration program especially developed for use in the OR. Each surgery can then use camera parameters computed *in situ* when the control structure was imaged shortly before surgery.

Feature Detection

1. For consistent success in detection of ball-features in images from one surgery to another, lighting and calibration (aperture and focal length) should be standardized. Currently, due to repeated calibration of cameras between surgeries, raw images have sometimes had to be manually preprocessed before images are input to DETECT-POINT. This is a problem with image acquisition and not

the method used for feature detection. Experimentation with various methods has shown that the feature detection problem in OR images is best solved by model-based recognition using cross-correlation, as formulated in DETECT-POINT.

2. Cross-correlation proved successful for feature detection in terms of accurate localization of features. However, correlation has its drawbacks when applied in a real-time environment because of its computational requirements. DETECT-POINT required an average of 240 seconds on an IBM RS/6000 to process a single OR image. It was calculated that the FFT implementation of cross-correlation formulated as part of this work will result in a 10-fold decrease in the time taken for correlation (24 sec/image). This includes multiple passes of the image with models of size 19x19 pixels. If the detection criteria are relaxed, so that the system does not depend on multiple models, there would be a 6-fold reduction in the gray level-space and a 4-fold reduction in the edge-space. This means that if only root templates of each property space were used with the FFT implementation of cross-correlation, the expected processing time would fall from 240 sec/image to approximately 5 sec/image (average of 4 and 6 sec/image). This modification will not affect the performance of detection as the models in any template tree are structurally very similar and can be approximated by the root template (pg. 64, tables 4.2, 4.3, for all proximity matrix elements, $\rho > 0.95$). This approach will lead to an increase in the number of false positives so that the onus of eliminating them would lie on consequent matching routines. However, as detection is computationally far more intensive than matching, this would be an acceptable compromise.

3. The triad-markers used had spherical components fabricated of plastic, painted black. Hard plastic and stainless steel were the only alternatives investigated as materials for the ball-components which were compatible with the OR. Though plastic has been used, it still presents lighting problems. For instance, ball features appear gray instead of black when overhead lighting is introduced to

eliminate shadows and light the 'hollow' of the back under surgery. A material with the properties of dull cloth, which does not glare, which is also easily admissible into a sterile environment would be ideal. Florescent paints and special illumination are possible alternatives which deserve experimentation. The use of color cameras for easier feature detection should also be experimented with. Colour image processing is still considered a relatively unexplored area [33]. It was not used because it increases the computational complexity of all algorithms. However, if a colour-based region segmentation approach provides results which are as accurate and computationally less complex than template matching, the extra expenditure in hardware (color cameras) may be justified.

4. For correlation, two property spaces, gray levels and edges, were used. *Directivity*, corresponding to the edge direction instead of magnitude, could be used as another property for correlation. Just as edges provide more information about the structure of a scene than gray levels, a study comparing directivity as an index of scene structure will prove useful.

Correspondence

1. The triad-marker configuration currently being used consists of a flat, planar arrangement of spherical ball-components and their unequal legs. This configuration reduces the problem of occlusion which commonly afflicted the earlier configuration where ball-components occupied the ends of three mutually orthogonal legs. The current 3D triad-model has been found to be 'geometrically sound' during 3D model matching. However, its design can be improved by solving for the location of ball-features in a plane as an optimization design problem where the distances between the ball features have to be mutually maximized. This could result in a smaller optimized model for triad markers.

2. Decision spaces, created as a result of correlation during feature detection, could be used by the intermediate stage of matching for the purpose of refinement of feature sets. This would speed up matching considerably as the area-based

correlation constraint could be implemented as a look-up into decision space instead. This was not implemented as the development of detection and matching routines were kept isolated from each other.

3. It will prove advantageous to future researchers working with this system to get a better idea of disparity distributions corresponding to the camera geometries they use. A control frame (*disparity grid*), of the size of the field of view of the cameras, can be constructed with points spanning the depth of field of the system in each region of the field of view. Digitization or auto-location of the disparity points of such a control frame could be done during calibration. An analysis of disparity over the field of view can provide important information which may be incorporated into the matching algorithm. For example, with a small baseline-vergence configuration, the disparity in the y-direction (in image frame of reference) was found to be within 14 ± 5 pixels. This could be used as an additional constraint during the refinement of candidate sets, the intermediate stage of the correspondence process.

Visualization

1. For this system to become clinically useful, triad measurements must be transformed to vertebral movements corresponding to the different stages of surgery. The previous chapter provides two alternatives for future work to do this. OpenGL can be used to get quantitative figures for the visual vertebral positions currently displayed by SPINE. This will not prove to be very tedious to implement. If the assessment of movements due to this approach are unsatisfactory, a photogrammetric basis will have to be implemented to relate stereo-camera and stereo-radiograph object spaces.

2. If data interpretation yields acceptable results, work can begin on the integration of the system into a self-contained, real-time platform for making intraoperative measurements.

7.3 Conclusions

A stereo-vision system was developed for the intraoperative monitoring of scoliosis surgery. The system was developed so that it can be integrated into a one-shot, non-invasive, on-line measurement tool used to intraoperatively record and quantitatively assess different stages of surgery .

The work done for the development of this system can be categorized into five areas:

- Mechanical System Design,
- Stereo-Camera Calibration,
- Feature Detection,
- Correspondence, and
- Visualization.

For each of these areas, a well-defined problem was formulated in view of the operating room environment, literature was surveyed, a solution was proposed and implemented. Groundwork was also laid for the system to be integrated into an easy-to-use automated tool. The approaches suggested [10] should remain the same unless major design changes are made. Development was slow because data collection and experimentation could only proceed within the constraints of real surgeries. As more OR data is collected, it should be possible to experiment with new materials for markers, repeatability of images, use of different baselines, and interpretation schemes. If real-time operation becomes crucial, the fast (using FFTs), relaxed (using root templates only) approach to detection, where the system currently spends more then 96% of its time, should be implemented. Quantitative interpretation of triad movements in terms of the vertebra they are rigidly attached to cannot be stressed enough. A visual interpretation is already available. It is these features which will endear the system to surgeons and clinicians.

Stereo-vision was used in the operating room for the monitoring of scoliosis correction. Currently, there are no widely used tools for this purpose. This work proves the feasibility of using stereophotogrammetry, computer vision, and image analysis in the OR. It lays the foundation for the automated, accurate 3D monitoring of scoliosis surgery using these methods. Based on my experience with this work, I can safely conclude that this approach, if pursued, will develop into a standard tool for the 3D intraoperative monitoring of scoliosis surgeries, among others.

Since the measuring device has been constructed by the observer ...
we have to remember that what we observe is not nature in itself
but nature exposed to our method of questioning

Physics and Philosophy (1958)
Werner Karl Heisenberg (1901-1976)

References

[1] Y.I. Abdel-Aziz and H. M. Karara. Direct linear transformation into object space coordinates in close-range photogrammetry. *Symposium on close-range photogrammetry, Univ. of Illinois, Urbana Champaign*, pages 1–18, 1971.

[2] J.K. Aggarwal, L.S. Davis, and W.N. Martin. Correspondence processes in dynamic scene analysis. *Proc. IEEE, 69, no.5*, pages 562–572, 1981.

[3] M. Asher. The Cobb angle should be used in the management of scoliosis. *Intl. Symp. on 3-D Scoliotic Deformities*, pages 57–58, 1992.

[4] D.H. Ballard and C.M. Brown. *Computer Vision.* Prentice-Hall, Inc., 1982.

[5] S.T. Barnard and M.A. Fischler. Computational stereo. *Computing Surveys, v.14, no.4*, pages 553–572, 1982.

[6] S.T. Barnard and W.B. Thompson. Disparity analysis of images. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, pages 333–340, 1980.

[7] B.G. Batchelor, D.A. Hill, and D.C. Hodgson. *Automated Visual Inspection.* IFS, 1985.

[8] K. Bhalla. *MATCH-POINT: An Algorithm for Point Feature Extraction and Registration in Stereoscopic Images.* Tech. Report (Course CMPUT 509) for Dr. A. Basu, Computing Science, University of Alberta, Canada, 1994.

[9] K. Bhalla. *Stereopsis: Some issues involving 3D measurements.* Tech. Report (Course EE 602) for Dr. N.G. Durdle, Electrical Engineering, University of Alberta, Canada, 1994.

[10] K. Bhalla, N.G. Durdle, A.E. Peterson, J. Raso, D. Hill, and X. Li. Automatic feature detection and correspondence in a stereo-vision application. *IEEE Conference on Systems, Man, and Cybernetics*, page *to be published*, 1995.

[11] M. Boninsegna and M. Rossi. Similarity measures in computer vision. *Pattern Recognition Letters*, pages 1255–1260, 1994.

[12] D. Braggins and J. Hollingum. *The Machine Vision Sourcebook.* IFS Publications, 1986.

[13] R.T. Chin and C.R. Dyer. Model-based recognition in robot vision. *ACM Computing Surveys, v.18, no.1*, pages 67–108, 1986.

[14] K. Chu. *Dynamic Visualization of the Spine using OpenGL.* EE602, Dept. of Electrical Engineering, University of Alberta, Canada, 1995.

[15] J.R. Cobb. Outline for the study of scoliosis. *Instructional Course Lectures, American Academy of Orthopaedic Surgeons*, pages 261–175, 1948.

[16] E.R. Davies. The median filter: an appraisal and a new truncated version. *Proc. 7th Intl. Conf. on Pattern Recognition*, pages 590–592, 1984.

[17] E.R. Davies. On the noise suppression and image enhancement characteristics of the median, truncated median and mode filters. *Pattern Recognition Letters*, pages 87–97, 1988.

[18] E.R. Davies. *Machine Vision: Theory, Algorithms, Practicalities.* Academic Press, 1990.

[19] U.R. Dhond and J.K. Aggarwal. Structure from stereo. *IEEE Trans. on Systems, Man, and Cybernetics, v.19, no.6*, pages 1489–1510, 1989.

[20] R.O. Duda and P.E. Hart. *Pattern Classification and Scene Analysis.* John Wiley & Sons, Inc., 1973.

[21] N.G. Durdle(*), J. Raso, and D. Hill. *Personal Communication.* (*)Dept. of Electrical Engg., University of Alberta; Rehabilitation Technology, Glenrose Hospital, Edmonton, Canada, 1994-95.

[22] O. Faugeras. *Three-Dimensional Computer Vision.* MIT Press, Cambridge, 1993.

[23] Ed. H. Freeman. *Machine Vision for Inspection and Measurement.* Academic Press, Inc., 1989.

[24] N.C. Gallagher and G.L. Wise. A theoretical analysis of the properties of median filters. *IEEE Trans. on Acoustics, Speech, and Signal Processing*, pages 1136–1141, 1981.

[25] G. De Giorgi, A. Gentile, and G. Mantriota. Three dimensional study of the spine: Our ten year experience. *Intl. Symp. on 3-D Scoliotic Deformities*, pages 71–80, 1992.

[26] C. A. Glasbey. An analysis of histogram-based thresholding algorithms. *CVGIP: Graphical Models and Image Processing, v.55, no.5*, pages 532–327, 1993.

[27] G.H. Gonnet. *Handbook of Algorithms and Data Structures.* Addison-Wesley, 1984.

[28] R.C. Gonzales and R.E. Woods. *Digital Image Processing.* Addison-Wesley Publishing Company, 1992.

[29] R.C. Gonzalez and P. Wintz. *Digital Image Processing.* Addison-Wesley Publishing Company, 1987.

[30] W.E.L. Grimson. Computational experiments with a feature based stereo algorithm. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, pages 17–34, 1985.

[31] S.S. Hu and R.S. Pashman. Spinal instrumentation. Evolution and state of the art. *Invest. Radiol.*, pages 632–647, 1992.

[32] X. Hu. *Perception of Shape and Motion.* Ph.D. thesis, Dept. of Electrical Engineering, Univ. of Illinois, Urbana Champaign, 1992.

[33] G.J. Klinker. *A Physical Apprach to Color Image Understanding.* A.K. Peters, Ltd., 1995.

[34] H. Labelle, J. Dansereau, C. Bellefleur, J. De Guise, C. Rivard, and I.A.F. Stokes. 3-D Correction obtained with the C-D procedure during surgery. *Intl. Symp. on Three Dimensional Analysis of Spinal Deformities*, pages 349–351, 1995.

[35] S.M. Lai, X. Li, and W.F. Bischof. Automated detection of breast tumors. *Vision Interface Conference*, pages 115–132, 1988.

[36] R.K. Lenz and R.Y. Tsai. Techniques for calibration of the scale factor and image center for high accuracy 3D machine vision metrology. *IEEE Trans. on Pattern Analysis and Machine Intelligence, v.10, no.5*, pages 68–75, 1987.

[37] X. Li. *Personal Communication.* Dept. of Computing Science, University of Alberta, Edmonton, Canada, 1995.

[38] X. Li, M. Ferdousi, M. Chen, and T.T. Nguyen. Matching with multiple templates. *Intl. Conf. on Computer Vision and Pattern Recognition*, pages 610–613, 1986.

[39] X. Li, C. Shanmugamani, T. Wu, and R. Madhavan. Correlation measures for corner detection. *Intl. Conf. on Computer Vision and Pattern Recognition*, pages 643–646, 1986.

[40] J. Mahood and M. Moreau. *Personal Communication.* Dept. of Surgery, University of Alberta Hospitals, Edmonton, Canada, 1994-95.

[41] D. Marr and T. Poggio. Cooperative computation of stereo disparity. *Science*, pages 283–287, 1976.

[42] L. Matthies. *Dynamic Stereo Vision.* Dept. of Computer Science, Carnegie Mellon Univ., Tech. Report CMU-CS-89-195, 1989.

[43] L. Matthies. *Personal Communication.* SRI International (formerly in Dept. of CS, Carnegie Mellon University), 1995.

[44] M. Mehta. The Cobb angle should not be used in the management of scoliosis. *Intl. Symp. on 3-D Scoliotic Deformities*, page 59, 1992.

[45] A. Mitiche and P. Bouthemy. Representation and tracking of point structures using stereovision. *Proceedings of the Workshop on Computer Vision: Representation and Control*, pages 118–124, 1984.

[46] F.H. Moffitt and E.M. Mikhail. *Photogrammetry.* Harper & Row, 1980.

[47] Hans P. Moravec. *Robot Rover Visual Navigation.* UMI Research Press, Michigan, 1981.

[48] R.T. Morrissy. *Atlas of Pediatric Orthopaedic Surgery.* J.B. Lippincott Company, 1992.

[49] F. Murtagh. A feature-based $O(n^2)$ approach to point pattern matching. *Intl. Conf. on Pattern Recognition*, pages 174–177, 1992.

[50] A. Peterson. *Personal Communication.* Dept. of Civil Engineering, University of Alberta, Edmonton, Canada, 1994-95.

[51] A. Peterson, N.G. Durdle, V.J. Raso, and D.L. Hill. Calibration of video cameras for scoliosis mapping. *Geomatica*, pages 47:29–38, 1993.

[52] S.B. Pollard, J.E.W. Mayhew, and J.P. Frisby. PMF: A stereo correspondence algorithm using a disparity gradient limit. *Perception*, pages 449–470, 1985.

[53] K.K. Pringle. *Visual Perception by a Computer, ed. A. Grasselli.* Academic Press, 1969.

[54] H.K. Ramapriyan. A multilevel approach to sequential detection of pictorial pictures. *IEEE Trans. on Computers, v.25, no.1*, 1976.

[55] S. Ranade and A. Rozenfeld. Point pattern matching by relaxation. *Pattern Recognition*, pages 269–275, 1980.

[56] G.G. Rusell, V.J. Raso, D. Hill, and J. McIvor. A comparison of four computerized methods for measuring vertebral rotation. *Spine, v.15, no.1*, pages 24–27, 1990.

[57] P.K. Sahoo, S. Soltani, A.K.C. Wong, and Y.C. Chen. A survey of thresholding techniques. *Computer Vision Graphics Image Processing, v.41*, pages 233–260, 1988.

[58] R. Schalkoff. *Pattern Recognition. Statistical, Structural, and Neural Approaches.* John Wiley & Sons, Inc., 1992.

[59] C.C. Slama(*Editor*). *Manual of Photogrammetry.* American Society of Photogrammetry, 1980.

[60] P. Suetens, P. Fua, and A.J. Hanson. Computational strategies for object recognition. *ACM Computing Surveys, v.24, no.1*, pages 5–61, 1992.

[61] S.J. Treadwell, K.S. Booth, C.W. Reilly, B.J. Sawatzky, and S.B. Jang. Three-dimensional analysis of scoliosis surgery using stereophotogrammetry. *Intl. Symp. on Three Dimensional Analysis of Spinal Deformities*, pages 95–100, 1995.

[62] R.Y. Tsai. A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses. *IEEE Journal of Robotics and Automation*, pages 323–343, 1987.

[63] B. Vandegriend, D. Hill, J. Raso, N.G. Durdle, and Z. Zhang. Application of computer graphics for the assessment of spinal deformities. *Medical & Biological Engineering & Computing*, pages 163–167, 1995.

[64] David Vernon. *Machine Vision - Automated Visual Inspection and Robot Vision.* Prentice Hall (UK), 1991.

[65] G.Q. Wei and S.D. Ma. Two plane camera calibration: A unified approach. *Proc. IEEE Conf. on Comp. Vision and Patt. Rec.*, pages 133–138, 1991.

[66] G.Q. Wei and S.D. Ma. A complete two-plane camera calibration method with experimental comparisons. *IEEE Intl. Conf. on Comp. Vision*, pages 439–446, 1993.

[67] Ed. S.L. Weinstein. *The Pediatric Spine: Principles and Practice.* Raven Press, 1994.

[68] J.S. Weszka and A. Rosenfeld. Histogram modification for threshold selection. *IEEE Transactions on Systems, Man, and Cybernetics, v.9, no.1*, pages 38–52, 1979.

[69] R. Wilson and M. Span. *Image Segmentation and Uncertainity.* Research Studies Press Ltd., 1988.

[70] P.R. Wolf. *Elements of Photogrammetry.* McGraw-Hill Book Company, 1983.

[71] P.R. Wolf. *Adjustment Computations: Practical Least Squares for Surveyors.* Landmark Enterprises, 1985.

[72] A.K.C. Wong. An algorithm for constellation matching. *Proc. of the 8th Intl. Conf. of Pattern Recognition*, pages 546–554, 1986.

Appendix A Correlation Speedup using FFTs

References to spatial domain correlation implemented in the frequency domain using the fast fourier transform (FFT) [29] usually refer to correlation in its orthodox form (OCC, eq. A.3). For correlation to be implemented in its more useful forms (RHO, CCM), there has been little work done to make these forms compatible for FFT implementations. This appendix presents the mathematical proof for the calculation of the normalized cross-correlation coefficient with means adjustment (eq. A.1) using fast fourier techniques. This form was chosen of the five investigated as all the others can be derived as subparts of this measure. Hence, their FFT forms can be easily obtained by selecting parts of the proof presented here.

For convenience, the form of the correlation coefficient being considered is restated in eq. A.1. The image, g(x,y), is of size MxM and the template, t(x,y), is of size NxN. As a simplification, the images are assumed to be square.

$$\rho(x,y) = \frac{\sum_{i,j=} \sum_{-\frac{N-1}{2}}^{\frac{N-1}{2}} \left(t(i,j) - \bar{t}\right)\left(g(x+i,y+j) - \bar{g}(x,y)\right)}{\sqrt{\left(\sum_{i,j=} \sum_{-\frac{N-1}{2}}^{\frac{N-1}{2}} \left(g(x+i,y+j) - \bar{g}(x,y)\right)^2\right)\left(\sum_{i,j=} \sum_{-\frac{N-1}{2}}^{\frac{N-1}{2}} \left(t(i,j) - \bar{t}\right)^2\right)}} \tag{A.1}$$

where $\bar{t}$ is the template average, computed once, and $\bar{g}(x,y)$ is the average of the NxN image shadow centered at (x,y).

$$\bar{t} = \frac{1}{N^2} \sum_{i,j=} \sum_{-\frac{N-1}{2}}^{\frac{N-1}{2}} t(i,j)$$

$$\bar{g}(x,y) = \frac{1}{N^2} \sum_{i,j=} \sum_{-\frac{N-1}{2}}^{\frac{N-1}{2}} g(x+i,y+j)$$

Eq. A.1 is a modified form of the basic cross-correlation equation given in eq. A.2. The derivation of eq. A.1 from eq. A.2 was dealt with earlier (pg. 51).

Lemma ℵ 1 *Given two functions, g() and t() (as described above), the convention regarding summation used in eq. A.2*

$$C(x,y) = \sum_{i,j=} \sum_{-\frac{N-1}{2}}^{\frac{N-1}{2}} t(i,j)g(x+i,y+j) \tag{A.2}$$

is equivalent to that followed in orthodox equations like eq. A.3

$$C(x,y) = \sum_{i,j=} \sum_{-\infty}^{\infty} g(i,j)t(i-x,j-y) \tag{A.3}$$

Proof

In eq. A.2, the limits are specifically shown in terms of N, the size of t(). However, because

$$t(i,j) = 0 \text{ , for i,j} \notin [-\tfrac{N-1}{2}, \tfrac{N-1}{2}]$$

eq. A.2 can be written as

$$C(x,y) = \sum_{i,j=} \sum_{-\infty}^{\infty} t(i,j)g(x+i,y+j)$$

Substituting

$$\alpha = x + i,$$

$$\beta = y + j$$

in the above form of eq. A.2, we get,

$$C(x,y) = \sum_{\alpha-x,\beta-y=} \sum_{-\infty}^{\infty} t(\alpha-x,\beta-y)g(\alpha,\beta)$$

or

$$C(x,y) = \sum_{\alpha,\beta=} \sum_{-\infty}^{\infty} g(\alpha,\beta)t(\alpha-x,\beta-y)$$

which is the same as eq. A.3.

QED

Theorem ℑ 1 (Convolution Theorem) *Given two functions* g *and* t *with Fourier transforms* G *and* T,

$$\mathcal{F}[\ g^{*}t(\mathrm{x,y})\ ] = \mathcal{F}[\ \mathrm{g}(x,y)\ ]\,\mathcal{F}[\ \mathrm{t}(x,y)\ ] = \mathrm{G}(f_x,f_y)\mathrm{T}(f_x,f_y)$$

This theorem has been stated for completeness only. For its proof, refer [20]. Next, the relation between correlation and convolution is shown so that correlation can be computed in the frequency domain using theorem 1.

Lemma ℵ 2 *Given two functions g(x,y) and t(x,y), correlation ($\bigotimes$)between the functions is equivalent to convolution (*) of the first function with the second flipped about both its axes, i.e.*

$$g \bigotimes t = g * \hat{t} \tag{A.4}$$

*where, $\hat{t} = t(-x,-y)$, the flipped function, and the operations $\bigotimes$ and * are defined as:*

$$\left(g \bigotimes f\right)(x,y) = \sum_{i,j=-\infty}^{\infty} g(i,j)t(i-x,j-y) \tag{A.5}$$

$$(g * f)(x,y) = \sum_{i,j=-\infty}^{\infty} g(i,j)t(x-i,y-j) \tag{A.6}$$

Proof

Expanding the RHS of eq. A.4 using the definition of convolution (eq. A.6), we get:

$$g * \hat{t} = \sum_{i,j=-\infty}^{\infty} g(i,j)\hat{t}(x-i,y-j) \tag{A.7}$$

but, it is known that

$$\hat{t}(x-i,y-j) = t(i-x,j-y)$$

Substituting this into eq. A.7, we have

$$g * \hat{t} = \sum_{i,j=-\infty}^{\infty} g(i,j)t(i-x,j-y) = g \bigotimes f$$

Hence, (A.4). *QED*

Corollary

$$g \bigotimes t = \mathrm{g} * \hat{t} = inv\mathcal{F}\left[\mathcal{F}\left(\mathrm{g}\right).\mathcal{F}\left(\hat{t}\right)\right]$$

Using the background just presented, eq. A.1 will be represented as multiple simplified terms, each of which is in the form of ordinary cross-correlation (eq. A.2) and can be evaluated using the fast Fourier transform by lemma 1. Consider eq. A.1 in the form,

$$\rho(x,y) = \frac{\mathcal{N}(\mathrm{x,y})}{\mathcal{D}(\mathrm{x,y})} \tag{A.8}$$

The template can undergo a one-time adjustment for its mean,

$$\tilde{t}(x,y) = t(x,y) - \bar{t}$$

The numerator (eq. A.8) can be expanded using eq. A.1,

$$\mathcal{N}(\mathrm{x,y}) = \sum_{i,j=-\frac{N-1}{2}}^{\frac{N-1}{2}} \tilde{t}(i,j)\,(g(x+i,y+j) - \bar{g}(x,y)) \tag{A.9}$$

$$= \mathcal{A}(\mathrm{x,y}) - \mathcal{B}(\mathrm{x,y}) \tag{A.10}$$

where,

$$\begin{aligned} \mathcal{A}(\mathrm{x,y}) &= \sum_i \sum_j \tilde{t}(i,j) g(x+i,y+j) \\ \mathcal{B}(\mathrm{x,y}) &= \sum_i \sum_j \tilde{t}(i,j) \bar{g}(x,y) \end{aligned} \tag{A.11}$$

The component term, $\mathcal{A}$(x,y), is in the simplified correlation form (eq. A.2) and can be evaluated using FFTs. Simplifying the term, $\mathcal{B}$,

$$\begin{aligned} \mathcal{B}(\mathrm{x,y}) &= \sum_i \sum_j \tilde{t}(i,j) \bar{g}(x,y) \\ &= \bar{g}(x,y) \sum_i \sum_j \tilde{t}(i,j) \qquad \text{as } \bar{g}(x,y) \text{ is independent of i,j} \\ &= \bar{g}(x,y).T^{SUM} \end{aligned} \tag{A.12}$$

where, $T^{SUM} = \sum_i \sum_j \tilde{t}(i,j)$, a constant for every template , and,

$$\bar{g}(x,y) = \frac{1}{N^2} \sum_{u,v=-\frac{N-1}{2}}^{\frac{N-1}{2}} g(x+u,y+v)$$

is the averaged image which can be computed (see lemma 3) using FFTs. This need be done only once and the value of $\bar{g}(x,y)$ will be available for each computation.

Lemma ℵ 3 *The image averaging operation is equivalent to cross-correlation.*

Proof

Consider the averaging operation,

$$\bar{g}(x,y) = \frac{1}{N^2} \sum_{u,v=-\frac{N-1}{2}}^{\frac{N-1}{2}} g(x+u,y+v)$$

The above can be rewritten as,

$$\bar{g}(x,y) = \sum_{u,v=} \sum_{-\frac{N-1}{2}}^{\frac{N-1}{2}} f(u,v)g(x+u,y+v) \text{ where, } f(x,y) = \tfrac{1}{N^2} \text{ for x,y} \in \left[-\tfrac{N-1}{2}, \tfrac{N-1}{2}\right]$$

QED

Consider the denominator, $\mathcal{D}()$, of eq. A.8. Squaring both sides,

$$\begin{aligned} [\mathcal{D}(\text{x,y})]^2 &= \left[\sum_i \sum_j \left[\tilde{t}(i,j)\right]^2\right] \left[\sum_i \sum_j [g(x+i,y+j) - \bar{g}(x,y)]^2\right] \quad \text{(A.13)} \\ &= \left[T^{SUMofSQUARES}\right] . [VAR(x,y)] \end{aligned}$$

The term, $T^{SUMofSQUARES} = \sum_i \sum_j \left[\tilde{t}(i,j)\right]^2$ is a constant for each template. The other term, VAR(x,y), is more involved as it represents the dynamic energy level of the shadow image relative to the average of the central pixel. Expanding this term,

$$\begin{aligned} VAR(x,y) &= \sum_i \sum_j [g(x+i,y+j) - \bar{g}(x,y)]^2 \quad \text{(A.14)} \\ &= \sum_i \sum_j \left[g(x+i,y+j)^2 + \bar{g}(x,y)^2 - 2\bar{g}(x,y)g(x+i,y+j)\right] \\ &= \mathcal{A}(\text{x,y}) + \mathcal{B}(\text{x,y}) - \mathcal{C}(\text{x,y}) \end{aligned}$$

(A.15)

Evaluating each component term of eq. A.15,

$$\begin{aligned} \mathcal{A}(\text{x,y}) &= \sum_i \sum_j \left[g(x+i,y+j)^2\right] \\ &= \sum_i \sum_j [f(i,j)gsq(x+i,y+j)] \\ &\quad \text{where, } gsq(x,y) = [g(x,y)]^2 \text{ and f(x,y)} = 1 \text{ for i,j} \in \left[-\tfrac{N-1}{2}, \tfrac{N-1}{2}\right] \end{aligned}$$

(A.16)

which is in the FFT-compatible correlation form.

$$\begin{aligned} \mathcal{B}(\text{x,y}) &= \sum_{i,j=} \sum_{-\frac{N-1}{2}}^{\frac{N-1}{2}} [\bar{g}(x,y)]^2 \quad \text{(A.17)} \\ &= N^2 [\bar{g}(x,y)]^2 \\ &= N^2 gAvSq(x,y) \text{ where, } gAvSq(x,y) = [\bar{g}(x,y)]^2 \end{aligned}$$

(A.18)

The $gAvSq$(x,y) image can be precomputed from the average image, $\bar{g}(x,y)$, as an $O(n^2)$ operation. We find that this image can be used in the simplified form of the final component, $\mathcal{C}()$, of the denominator too.

$$\begin{aligned}
\mathcal{C}(\text{x,y}) &= \sum_i \sum_j 2\bar{g}(x,y)g(x+i,y+j) \\
&= 2\bar{g}(x,y)\left[\sum_i \sum_j g(x+i,y+j)\right] \qquad \text{as } \bar{g}(x,y) \text{ is independent of i,j} \\
&= 2\bar{g}(x,y)\left[N^2\bar{g}(x,y)\right] \qquad \text{by the defn. of the image average} \\
&= 2N^2\left[\bar{g}(x,y)\right]^2 \qquad \text{(A.19)} \\
&= 2N^2 gAvSq(x,y) \qquad \text{(A.20)}
\end{aligned}$$

The final components, $\mathcal{B}()$ and $\mathcal{C}()$, of eq. A.13 can be simplified as

$$\begin{aligned}
\mathcal{B}(\text{x,y}) - \mathcal{C}(\text{x,y}) &= N^2\left[\bar{g}(x,y)\right]^2 - 2N^2\left[\bar{g}(x,y)\right]^2 \\
&= -N^2\left[\bar{g}(x,y)\right]^2 \\
&= -N^2 gAvSq(x,y)
\end{aligned} \qquad \text{(A.21)}$$

Eq. A.1 is thus simplified into (x,y)-position dependent components rather than neighbourhoods. Each of these components can be computed using the basic form of cross-correlation (eq. A.2) or simply referenced from a precomputed image (like gAvSq(x,y)) or template-dependent constant (like T^{SUM}). This formulation can prove extremely efficient when normalized cross-correlation involves templates of large sizes.

Appendix B System Specifications

This appendix briefly describes system specifications in terms of the software developed and platform used.

Software Information

Programming Language : ANSI C

Operating System : UNIX

Compiler : gcc (for portability to different platforms)

Platform : IBM RS/6000

Graphics Support : OpenGL (Hardware assisted on the IBM RS/6000)

Software Available With : Dr. N.G. Durdle, Dept. of Electrical Engineering, University of Alberta, Edmonton. Alberta. Canada T6G 2G7.

Calibration Software

Directory Name : Calibration

Executable Names : pix2mm, ndlt, v2stereo

Detection Software

Directory Name : DetectPoint

Executable Name : detect

Image format supported : SUN Rasterfile

Correspondence Software

Directory Name : MatchPoint

Executable Name : match

Hardware Information

Refer table 3.1

Appendix C Software Model: DETECT-POINT

This appendix outlines the software model for DETECT-POINT, the software developed for the automatic extraction of features in OR images. The relationship between components of the software (modules) is presented using the C files used to build DETECT-POINT. Control dependencies between the modules is presented graphically (fig. C.1). The functionality of each of the modules are given with the relevant inputs and outputs. Code is heavily documented and contains all other information necessary to understand or modify the software.

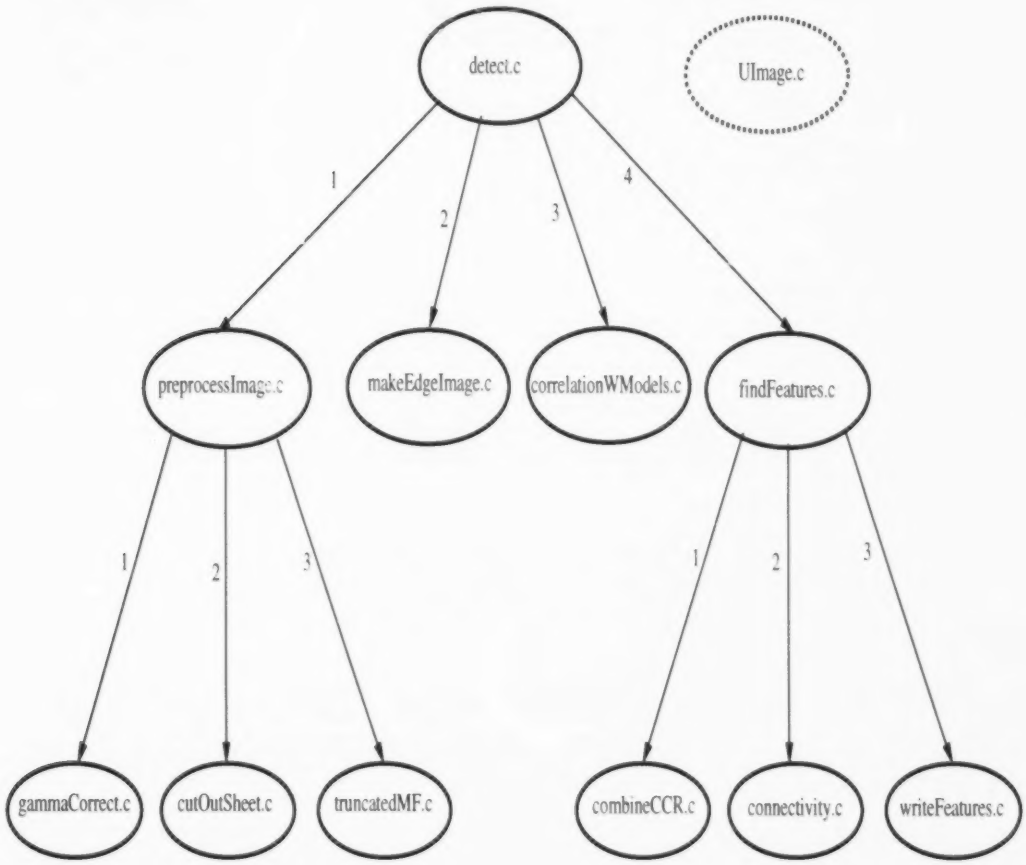

Figure C.1: *DETECT-POINT: Software Modules.*

C.1 Module Descriptions

The modules comprising DETECT-POINT are shown in fig. C.1. Numbers indicate the chronological sequence in which modules are accessed from their parent control nodes. All modules are independent. Control is always returned to the parent module, unless specified otherwise. A high level description is provided here of each of the modules in terms of its purpose, reference (theory) in the thesis, input, and output. Details from within modules like individual functions, data structures, and variables are provided in the introductory comments of module code and the README files written for future programmers.

detect.c This is the control module for DETECT-POINT. It is only responsible for passing program control to the four main modules of DETECT-POINT (preprocessImage.c, makeEdgeImage.c, correlationWithModels.c, findFeatures.c) in a specified order. After each of these modules return control to detect.c, results of that stage are recorded in images (pre.ras, edge.ras, crossGl.ras crossEd.ras, final.ras) for data analysis and debugging purposes.

Input Operating room image in SUN Rasterfile format.

Output Data file (FEATURES.DAT) containing the number of features detected and the list of features and their subpixel locations.

preprocessImage.c This module is responsible for the preprocessing of images. It sequentially calls modules gammaCorrection.c, cutOutSheet.c, and truncated-MedianFilter.c in that order.

Input Raw image.

Output Preprocessed image (after gamma correction, size reduction, and TMF).

gammaCorrection.c This module contains code for the gamma correction algorithm.

Reference pg. 33.

Input Raw image.

Output Raw image after gamma correction.

cutOutSheet.c This module contains code for the automatic size reduction algorithm.

Reference pg. 34.

Input Image after gamma correction.

Output Image after size reduction.

truncatedMedianFilter.c This module contains code for the implementation of the truncated median filter.

Reference pg. 37.

Input Image after size reduction.

Output Image after truncated median filtering.

makeEdgeImage.c This module contains code for the implementation of the 3x3 Sobel Edge detector.

Reference pg. 45.

Input Preprocessed image data structure after TMF.

Output Edge image corresponding to the prepocessed image.

correlationWithModels.c This module contains code for the implementation of cross-correlation using gray level or edge templates, according to the input image.

Reference pg. 48.

Input Gray level (preprocessed) image or edge image and flag indicating property space.

Output Correlation image (decision space) containing cross-correlation values (ρ) scaled to 0-255.

findFeatures.c This module controls the task of feature extraction. It sequentially calls modules combineCCResults.c, connectivity.c, and writeFeatures.c in that order.

Input Correlation images containing gray level and edge decision spaces.

Output Data file (FEATURES.DAT) containing list of features and their loca-

tions.

combineCCResults.c This module combines the results of cross-correlation using gray levels and edges to form the feature space.

Reference pg. 66.

Input Correlation images containing gray level and edge decision spaces.

Output Feature space image after thresholding.

connectivity.c This module implements the 8-connectivity algorithm to isolate feature components after thresholding of the feature space.

Reference pg. 66.

Input Feature space image after thresholding.

Output List of 8-connected components and their sub-pixel centroid locations.

writeFeatures.c This module writes the feature set determined to a data file.

Input List of 8-connected components and their sub-pixel centroid locations.

Output Data file (FEATURES.DAT) containing list of features and their locations.

UImage.c While the other modules are independent, this module contains a library of global functions. The commercial SUN Rasterfile format for images is supported by DETECT-POINT. Internally, images are maintained in a data structure. In all the above statements, images implicitly mean image data structures. Functions in this module are called universally for reading (SUN Rasterfile) images into image data stuctures and for writing from image data structures into (SUN Rasterfile) images.

Input SUN Rasterfile image or image data structure.

Output Image data structure or SUN Rasterfile respectively.

Appendix D Software Model: MATCH-POINT

This appendix outlines the software model for MATCH-POINT, the software developed for the automatic registration of feature sets corresponding to stereo-images. The relationship between components of the software (modules) is presented using the C files used to build MATCH-POINT. Control dependencies between modules is presented graphically (fig. D.1). The functionality of each of the files is given with the relavent inputs and outputs. Code is heavily documented and contains all other information necessary to understand or modify the software.

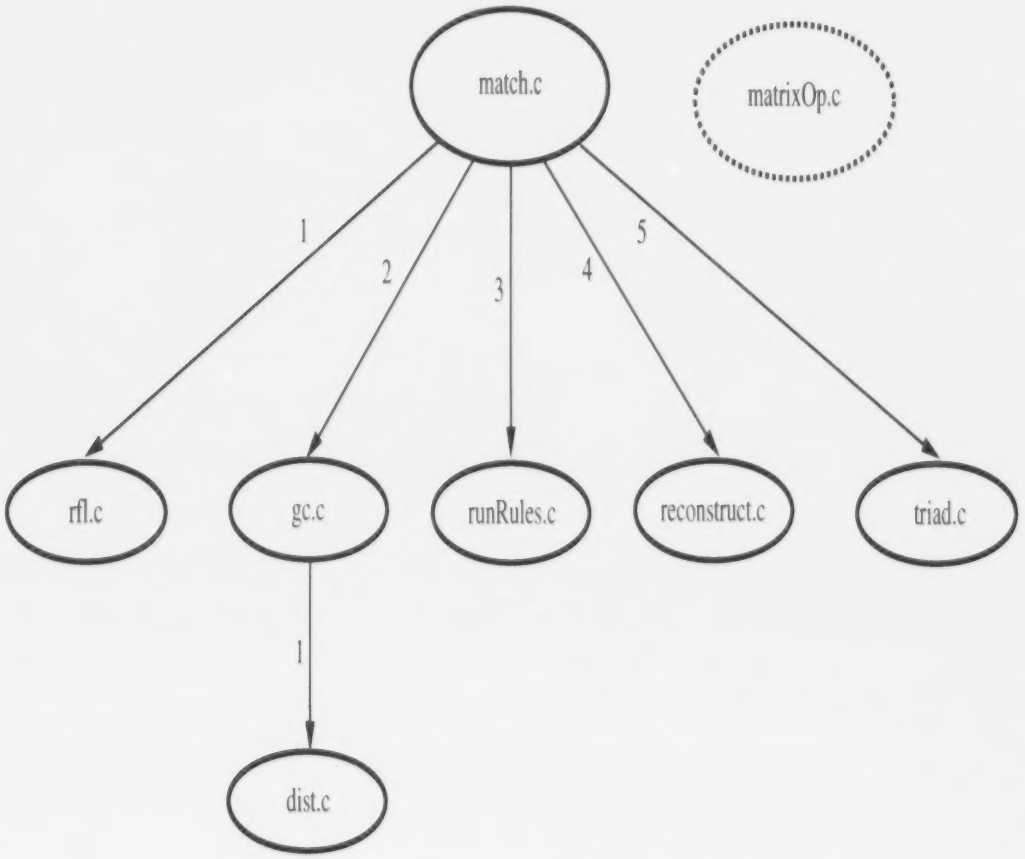

Figure D.1: *MATCH-POINT: Software Modules.*

D.1 Module Descriptions

The modules comprising MATCH-POINT are shown in fig. D.1. The numbers indicate the chronological sequence in which modules are accessed from their parent control node. Control is always returned to the parent module, unless specified otherwise. A high level description is provided here of each of the modules in terms of its purpose, reference (theory) in the thesis, input, and output. Details from within modules like individual functions, data structures, and variables are provided in the introductory comments of module code and the README files written for future programmers.

match.c This is the control module for MATCH-POINT. It is only responsible for passing program control to the five main modules of MATCH-POINT (rfl.c, gc.c, runRules.c, reconstruct.c, triad.c) in a specified order.

Input Data files containing lists of features and their locations in both stereo-images.

Output Data file containing labelled triads, their object-space positions and orientations.

rfl.c rfl stands for read-feature-list. This module contains code for reading the feature sets for the stereo-images and the calibration parameters of the geometry with which these images were acquired. Pixel features are converted to millimetres and corrected for lens distortion. Extrinsic calibration parameters are used for to compute rotation and translation matrices used by other modules.

Reference pg. 79.

Input Data files containing feature pixels (PixelsLEFT.dat, PixelsRIGHT.dat) and stereo-camera parameters (CameraParameters.dat).

Output Data files and lists containing features plate coordinates in mm (corrected for lens distortion) and data structures containing rotation and translation matrices relating the left and right cameras.

gc.c gc stands for get-candidates. This module implements the lower levels of the correspondence hierarchy used to construct candidate sets for all features in

either image. It calls the module dist.c.

Reference pg. 79.

Input The *from* list of features, the *to* list of features, and flags relating from/to lists to left/right feature sets.

Output Data structures for all features containing their candidate sets.

dist.c The epipolar constraint used by gc.c is implemented here.

Reference pg. 81.

Input Feature (*from* feature) point structure, its close and far point projections, and *to* list of features.

Output Suitable candidates for the (*from*) feature chosen from the *to* list of features.

runRules.c This module contains rules for refinement of candidate sets. It can be disabled to speed up the matching process.

Input Feature structures from both images containing candidate sets.

Output Feature structures from both images containing refined candidate sets and data file (MatchCoords.dat) of points for reconstruction in input format of CSPACO.

reconstruct.c This module contains CSPACO, the C adaptation of the FORTRAN routine SPACO. It is responsible for returning 3D coordinates for points whose locations are specified in both images.

Reference pg. 87.

Input Data files containing camera parameters (CameraParameters.dat) and image points for reconstruction (MatchCoords.dat).

Output File containing 3D object-space coordinates of constructed points (World-Points.dat) and residuals data (SpacoOUT.dat) about reconstructed points for analysis and future work.

triad.c This module implements the highest tier of the correspondence process. It deals entirely with the 3D point feature set reconstructed in the module recon-

struct.c. Three dimensional model matching and triad evaluation by heuristics is implemented here.

Reference pg. 89.

Input List of 3D object-space points.

Output Data file (SpineStageX.dat) containing triads, their position, and orientation information.

matrixOp.c This module contains libraries for the matrix operations involved in modules like rfl.c.

Input Matrix structures and matrix operation expected.

Output Matrix structure corresponding to output of matrix operation.